



# **NAVAL POSTGRADUATE SCHOOL**

**MONTEREY, CALIFORNIA**

## **THESIS**

**MALICIOUS ACTIVITYSIMULATION TOOL (MAST) AND  
TRUST**

by

Brian J. Diana

June 2015

Thesis Advisor:  
Co-Advisor  
Second Reader

Karen Burke  
John Gibson  
Gurminder Singh

**Approved for public release; distribution is unlimited**

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE June 2015		3. REPORT TYPE AND DATES COVERED Master's Thesis
4. TITLE AND SUBTITLE MALICIOUS ACTIVITYSIMULATION TOOL (MAST) AND TRUST			5. FUNDING NUMBERS	
6. AUTHOR(S) Brian J. Diana				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____ N/A ____.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words)  Malicious Activity Simulation Tool (MAST) provides an on-the-job-training medium for information system operators to practice responding to cyber threats simulated on the operational information systems that they manage day-to-day. Because MAST has the capability to simulate various cyber attacks, it is important to measure the risk the system poses to the information systems on which it will operate.  This thesis analyzes MAST's security posture and proposes potential solutions to any vulnerability discovered in that analysis. The analysis is based on a Security Control Assessment (SCA) utilizing the Defense Information Systems Agency Application Security and Development Security Technical Implementation Guide. Following the SCA, a threat model is used to determine mitigations to technical findings. Outputs from this research will enable a more secure implementation of MAST.				
14. SUBJECT TERMS Malware, network security, training, Security Control Assessment, threat model			15. NUMBER OF PAGES 113	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release; distribution is unlimited**

**MALICIOUS ACTIVITYSIMULATION TOOL (MAST) AND TRUST**

Brian J. Diana  
Civilian, Department of the Navy  
B.S., East Stroudsburg University, 2006

Submitted in partial fulfillment of the  
requirements for the degree of

**MASTER OF SCIENCE IN COMPUTER SCIENCE**

from the

**NAVAL POSTGRADUATE SCHOOL  
June 2015**

Author: Brian J. Diana

Approved by: Karen Burke  
Thesis Advisor

John Gibson  
Co-Advisor

Gurminder Singh, Ph.D.  
Second Reader

Peter J. Denning, Ph.D.  
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

## **ABSTRACT**

Malicious Activity Simulation Tool (MAST) provides an on-the-job-training medium for information system operators to practice responding to cyber threats simulated on the operational information systems that they manage day-to-day. Because MAST has the capability to simulate various cyber attacks, it is important to measure the risk the system poses to the information systems on which it will operate.

This thesis analyzes MAST's security posture and proposes potential solutions to any vulnerability discovered in that analysis. The analysis is based on a Security Control Assessment (SCA) utilizing the Defense Information Systems Agency Application Security and Development Security Technical Implementation Guide. Following the SCA, a threat model is used to determine mitigations to technical findings. Outputs from this research will enable a more secure implementation of MAST.

THIS PAGE INTENTIONALLY LEFT BLANK



## TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	OBJECTIVES.....	1
B.	METHODOLOGY.....	2
C.	BENEFITS OF THIS RESEARCH TO THE DEPARTMENT OF DEFENSE AND DEPARTMENT OF THE NAVY .....	2
D.	THESIS ORGANIZATION.....	2
II.	BACKGROUND.....	5
A.	MAST ARCHITECTURE .....	5
1.	MAST Scenario Generation Server .....	6
2.	MAST Scenario Execution Server .....	7
3.	MAST Client.....	7
4.	MAST Simware.....	8
5.	Scenario Files .....	8
B.	TRUST .....	8
C.	RISK MANAGEMENT FRAMEWORK.....	9
D.	SECURITY CONTROL ASSESSMENT .....	11
III.	SECURITY CONTROL ASSESSMENT APPROACH .....	13
A.	PROGRAM MANAGEMENT CONSIDERATIONS .....	13
B.	DESIGN AND DEVELOPMENT CONSIDERATIONS .....	14
C.	SOFTWARE CONFIGURATION MANAGEMENT CONSIDERATIONS .....	19
D.	TESTING CONSIDERATIONS .....	19
E.	DEPLOYMENT CONSIDERATIONS .....	20
F.	SUMMARY .....	23
IV.	ASSESSMENT FINDINGS .....	25
A.	ASSESSMENT ORGANIZATION .....	25
B.	OVERVIEW OF FINDINGS .....	26
C.	PROGRAM MANAGEMENT FINDINGS.....	27
D.	DESIGN AND DEVELOPMENT FINDINGS.....	30
E.	SOFTWARE CONFIGURATION MANAGEMENT FINDINGS.....	38
F.	TESTING FINDINGS.....	40
G.	DEPLOYMENT FINDINGS .....	42
H.	SUMMARY .....	48
V.	MITIGATION AND REMEDIATION THROUGH THREAT MODELING.....	49
A.	THREAT MODEL BASICS .....	50
B.	MAST USE SCENARIOS.....	50
C.	MAST EXTERNAL DEPENDENCIES.....	50
D.	MAST SECURITY ASSUMPTIONS .....	51
E.	MAST DATA FLOW DIAGRAMS .....	51
F.	MAST THREAT TYPES .....	54

G.	IDENTIFY THREATS TO MAST .....	55
H.	DETERMINE RISK TO MAST.....	57
I.	PLAN MITIGATIONS FOR MAST.....	58
J.	THREAT MODEL LIMITATIONS .....	60
K.	SUMMARY .....	70
VI.	CONCLUSION AND FUTURE WORK .....	71
A.	CONCLUSION .....	71
B.	FUTURE WORK.....	72
1.	Application Tier-to-Tier Trust .....	72
2.	Simware Trust.....	72
3.	Additional SCA.....	72
	APPENDIX. DISA APPLICATION SECURITY STIG AREAS.....	73
	LIST OF REFERENCES.....	93
	INITIAL DISTRIBUTION LIST .....	95

## LIST OF FIGURES

Figure 1.	Simplified MAST Architecture.....	6
Figure 2.	MAST DFD with MAST Components.....	52
Figure 3.	MAST Simware and Scenario File Distribution.....	53
Figure 4.	MAST DFD with User .....	54

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF TABLES

Table 1.	DISA Severities, after [1] .....	25
Table 2.	STIG Check Status.....	26
Table 3.	Overall Summary of Findings .....	26
Table 4.	Summary of Program Management Findings .....	27
Table 5.	Program Management Findings Details, after [1] .....	27
Table 6.	Summary of Design and Development Findings .....	30
Table 7.	Design and Development Findings Details, after [1].....	31
Table 8.	Summary of Software Configuration Management Findings .....	38
Table 9.	Software Configuration Management Findings Details, after [1].....	39
Table 10.	Summary of Testing Findings.....	40
Table 11.	Testing Findings Details, after [1] .....	40
Table 12.	Summary of Deployment Findings.....	42
Table 13.	Deployment Findings Details, after [1].....	43
Table 14.	DFD Item Inventory .....	55
Table 15.	Reduced DFD Item Inventory .....	56
Table 16.	Threats to MAST .....	56
Table 17.	Mitigations Not Found in Threat Model, after [1].....	60
Table 18.	DISA Application Security STIG Areas, after [1] .....	73

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF ACRONYMS AND ABBREVIATIONS

ASD	Application Security and Development
ATO	Authorization to Operate
C&A	Certification and Accreditation
C2	Command and Control
CCB	Configuration Control Board
DFD	Dataflow Diagram
DIACAP	DOD Information Assurance Certification and Accreditation Process
DISA	Defense Information Systems Agency
DMZ	Demilitarized Zone
DOD	Department of Defense
DoN	Department of the Navy
DoDIN	Department of Defense Information Network
DoS	Denial of Service
IA	Information Assurance
IP	Internet Protocol
IS	Information System
ISSM	Information System Security Manager
IT	Information Technology
MAC	Media Access Control
MAC	Mission Assurance Category
MAST	Malicious Activity Simulation Tool
NIST	National Institute of Standards and Technology
OWASP	Open Web Application Security Project

PKI	Public Key Infrastructure
POAM	Plan of Actions and Milestones
PPS	Ports, Protocols, and Services
RMF	Risk Management Framework
SCA	Security Control Assessment
SDLC	Software Development Life Cycle
SE	Scenario Execution
SG	Scenario Generation
SSP	System Security Plan
STIG	Security Technical Implementation Guide



## **ACKNOWLEDGMENTS**

This thesis would not have been possible without the guidance, patience, and support of my thesis advisors, Professors Karen Burke and John Gibson, and Dr. Gurminder Singh. Each of you exhausted great personal energy to get me through this project, all the while from the other side of the continent. Thank you for your time and the opportunity to work with you.

A very special acknowledgement goes to my girlfriend Kelly Lane who loved and supported me through this whole endeavor. She kept me on target and encouraged me.

THIS PAGE INTENTIONALLY LEFT BLANK

# **I. INTRODUCTION**

Cybersecurity continues to be one of the most difficult challenges for the Department of Defense (DOD). Adversary tactics are always evolving, as are the information technology environments that they attack. Operators remain challenged to keep up with this tumult and must undergo constant training in order to remain competitive with their adversaries. Existing platforms, such as classroom training and various web- and computer-based training, offer a baseline of educational opportunity. However, they often do not provide experiences immediately and directly applicable to the environments operators regularly manage.

Malicious Activity Simulation Tool (MAST) provides a unique opportunity for information system operators to seize on-the-job-training and practice responding to cyber threats simulated on the operational information systems that they manage day-to-day. MAST is a software suite that allows for simulated malware activity on operational platforms. It creates the opportunity for practical training of defending the information systems that operators are charged to operate, manage and protect.

Before the MAST opportunity can be seized, it is important to measure the risk the system poses to the operational information systems (IS) that it will support. Particularly, due to MAST's capability to simulate various cyber attacks, increased rigor must be exercised in its assessment. This need for specialized training is the driver for this thesis.

## **A. OBJECTIVES**

The objective of this research is to analyze the security MAST's security posture and propose potential solutions to any vulnerabilities and findings that are identified. Output from this research will form a roadmap for program managers, developers, and user representatives in preparing MAST for an

authority to operate (ATO) on operational IT platforms. An ATO is required before MAST can be fielded on any operational platform.

## **B. METHODOLOGY**

The assessment contained in this thesis follows the process discussed in [1]. Particular emphasis is on the unique risk MAST represents as a malware simulation platform designed to run on production and operational platforms such as ship or installation networks. Greater detail about how the analysis is conducted is discussed in Chapter III, Security Analysis Approach.

## **C. BENEFITS OF THIS RESEARCH TO THE DEPARTMENT OF DEFENSE AND DEPARTMENT OF THE NAVY**

As previously stated, MAST is unique in both the mission it serves to accomplish and the risk it represents to the platforms and operators it would support. By conducting this assessment, we can create and then execute a roadmap to mitigate or eliminate the risks presented by MAST. This will pave the way for deployment of MAST onto its target networks and facilitate richer and more effective training for cyber operators.

## **D. THESIS ORGANIZATION**

This thesis is organized into several chapters, first introducing the research and the MAST system and then documenting the approach for its analysis, its findings, and ways to address those findings. It concludes with a summary of the analysis and what steps should follow this research. Further details about the chapters are as follows.

Chapter I introduces the research, its objectives, and importance. Chapter II provides an overview of MAST as well as prior research on the program. It also provides a brief overview of a security analysis. Chapter III details specific activities, resources, tools, and techniques used in the security analysis of MAST. Chapter IV documents specific findings and vulnerabilities discovered in the security analysis. Chapter V presents techniques and recommendations for

addressing the vulnerabilities discovered in the security analysis. Chapter VI provides an executive summary of MAST's security vulnerabilities and possible methods to address those findings as documented in Chapters IV and V.

THIS PAGE INTENTIONALLY LEFT BLANK

## **II. BACKGROUND**

This chapter describes the architecture and usage of MAST and also describes the intended focus areas for the security analysis. [2], [3], and [4] each go into significant detail about MAST architecture, so only cursory coverage will be provided here.

### **A. MAST ARCHITECTURE**

The MAST architecture is a three-tiered client server architecture, as illustrated in Figure 1. It features two server tiers and a client tier: the MAST Scenario Generation (SG) Server, the MAST Scenario Execution (SE) Server, and the MAST Clients.

The architecture also utilizes two sets of files: client module files, referred to as Simware, and scenario files. Both sets are used to guide and execute scenarios for MAST.

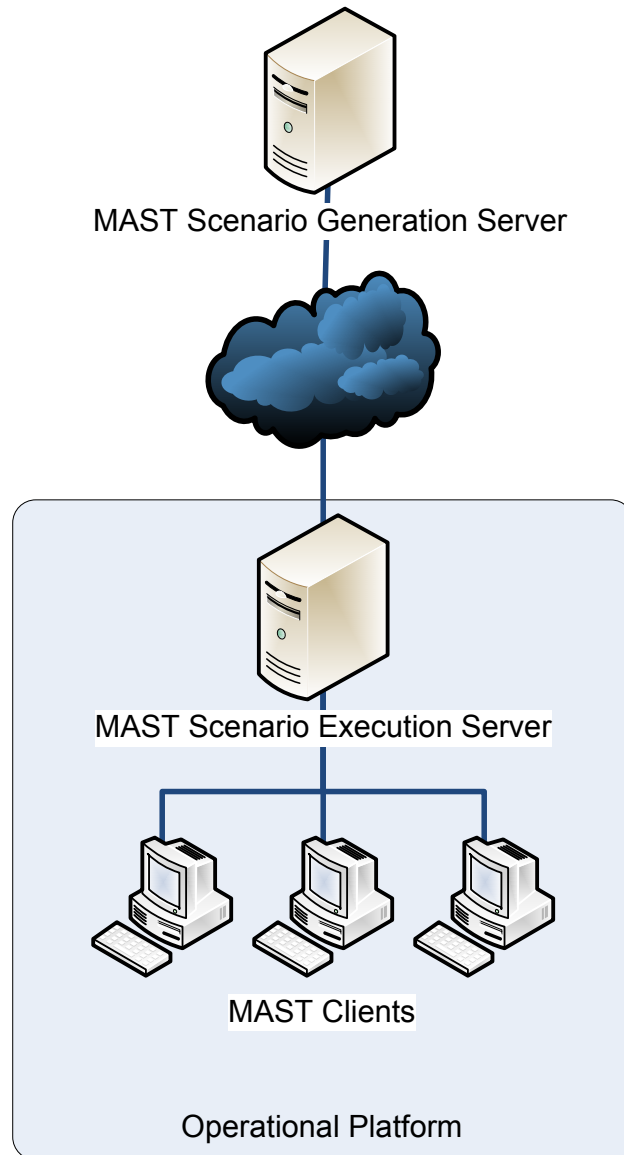


Figure 1. Simplified MAST Architecture

## 1. **MAST Scenario Generation Server**

The SG Server is the highest tier in the system's architecture. It installs as a simple Java-based desktop application. It provides the following capabilities and functions:

- Maintains a master library of Simware and Scenario files
- Deploys Simware and Scenario files to subordinate MAST SE Servers



- Provides for remote scenario management
- Generates reports and aggregates data from subordinate MAST SE

## **2. MAST Scenario Execution Server**

The SE Server is the primary controller for a scenario at an operational platform. It is also the first MAST product to be deployed on an operation platform, as opposed to the SG Server, which need not be deployed on the operational platform and thus introduces a different risk to the operators' mission. It provides the following capabilities and functions:

- Maintains a local library of Simware and Scenario files
- Deploys Simware and Scenario files to subordinate MAST Clients
- Receives status and logs data from MAST Clients
- Generates reports and data aggregates data from subordinate MAST Clients
- Maintains MAST client and Simware status
- Commands and controls all scenarios and MAST Clients
- Performs "kill switch" functionality allowing for emergency shutdown of a scenario

## **3. MAST Client**

The MAST Client is installed on each endpoint workstation running a Microsoft Windows operating system. It provides the following capabilities and functions:

- Runs Simware as directed by MAST SE
- Inventories host attributes, such as installed software and profiling information, such as Internet Protocol (IP) and Media Access Control (MAC) Addresses
- Reports status on scenario to MAST SE

#### **4. MAST Simware**

MAST Simware provides the muscle for each client to perform various malware simulations, including—but not necessarily limited to—port scans, antivirus triggers, and rogue pop-up advertisements. Requirements for Simware and each module’s capabilities and functions will vary widely and are often tailored for specific scenarios.

#### **5. Scenario Files**

MAST Scenario Files are intended to allow for pre-built and template scenarios to be issued from the SG Server. While not actual code, they script execution of MAST Simware across multiple clients.

### **B. TRUST**

In this paper we use trust to encompass the ability for various parts of the MAST architecture, including operators and developers, to ensure confidentiality, integrity, and availability as protected throughout MAST and the operation of MAST.

Confidentiality, integrity, and availability are the measures by which cybersecurity is assessed. The quality that security controls persevere in these areas can demonstrate how well an application or system protects itself, its data, and its users. Brief definitions of each of the three focus areas are as follows, as provided in [5]:

- Confidentiality—The property that information is not disclosed to system entities (users, processes, devices) unless they have been authorized to access the information.
- Integrity—The property whereby an entity has not been modified in an unauthorized manner.
- Availability—The property of being accessible and usable upon demand by an authorized entity.

Trust, as it is referred to in this paper, roughly equates to the trust as a concept in [6]; trust is a property of an entity to do as it is specified. When measuring trust, we refer to an object's trustworthiness.

Trustworthiness of information systems such as MAST is measured by how the information system uses defenses—known as security controls—to defend itself against potential environmental and human driven threats, ranging from malfunction to malicious. A security control assessment evaluates the adequacy in which the security controls reduce risk.

### **C. RISK MANAGEMENT FRAMEWORK**

A security control assessment produces the results and artifacts that are consumed in the Risk Management Framework (RMF) as documented in [7]. RMF replaces the Department of Defense Information Assurance Certification and Accreditation Process (DIACAP). The DOD RMF is better honed to build security functionality and management processes early into an information system's life cycle and incorporate continuous monitoring of a system's posture and risk well after its initial authorization to operate.

The Risk Management Framework (RMF) consists of six steps. These steps are not necessarily required to be in this order, however, as the RMF was designed to align to a development life cycle this is the ideal order. It should be noted that RMF may be applied to a legacy system that is already fielded however some gap analysis and assessment may be required.

The RMF steps are briefly discussed in the following sections. Since the focus of this thesis isn't to analyze RMF, only a cursory discussion will be included. Refer to [7] for further details.

(1) Step 1: Categorize Information System

In this step the system's threats and the information types it processes are determined.

(2) Step 2: Select Security Controls

In this step the security controls that will be implemented to address system threats and information types it processes are determined. The system security plan and continuous monitoring strategy are also finalized.

(3) Step 3: Implement Security Controls

In this step security controls are implemented and deployed. Supporting documentation is also updated.

(4) Step 4: Assess Security Controls

In this step a security control assessment is conducted against the information system. The output is equivalent to the "certification" task of traditional C&A.

(5) Step 5: Authorize Information System

In this step, a Plan of Action and Milestones (POAM) is created based on findings from Step 4. A POAM and other artifacts are bundled together in a security authorization package, along with a risk determination and presented to an Authorizing Official for risk acceptance. This is equivalent to the "accreditation" task of traditional C&A.

(6) Step 6: Monitor Security Controls

In this step typical system maintenance occurs to include configuration management and vulnerability management. Outstanding POAM items are also addressed.

#### **D. SECURITY CONTROL ASSESSMENT**

As defined in The Committee on National Security Systems National Information Assurance (IA) Glossary [5], a security control assessment (SCA) is:

The testing and/or evaluation of the management, operational, and technical security controls to determine the extent to which the controls are implemented correctly, operating as intended, and producing the desired outcome with respect to meeting the security requirements for the system or enterprise.

The practice of conducting a security control assessment is typically a critical step in acquiring authorization to operate an information system. For example, [8] requires that IT Product Information System Security Managers (ISSM) ensure their system undergoes a SCA.

The SCA conducted on MAST as part of the research for this thesis comprises the [1]. Further details on how the STIG is applied to MAST will be discussed in Chapter III, Security Control Assessment Approach.

THIS PAGE INTENTIONALLY LEFT BLANK

### **III. SECURITY CONTROL ASSESSMENT APPROACH**

The ASD STIG [1] remains the core component of the SCA conducted in this research. It represents significant research in an application development life cycle within the DOD, drawing on knowledge and capability from resources such as the National Institute of Standards and Technology, the Microsoft Corporation, the MITRE Corporation, and the Open Web Application Security Project (OWASP) [1].

The ASD STIG [1] divides an SCA into five areas built around a Software Development Life Cycle (SDLC). The advantage to this approach is that application developers can advance with greater ease through development and implementation of the STIG as they develop their application.

Areas in [1] that are not applicable to MAST are omitted from discussion in this chapter. For example, security concerns specific to web technologies will not be discussed as those technologies are not in use by MAST.

#### **A. PROGRAM MANAGEMENT CONSIDERATIONS**

Program management focuses on planning functions of an application's development. Of the 158 tests in [1], 16 emphasize program management. Analysis steps in this area focus on system documentation and planning processes, as well as the beginning stages of application deployment. Interestingly, half of the checks that apply to program management also apply to deployment.

##### **(1) Documentation**

The first stage of reviewing for program management considerations includes documentation review. Particular artifacts to assess include:

- System Security Plan (SSP)—A keystone artifact that documents all security aspects of a program. It is often a primary artifact in a package forwarded to an authorizing official when applying for an authority to operate.

- Application Configuration Guide—A document that captures how an application is to be deployed in its target environment. It often also includes information on securing the application in accordance with the controls and capabilities documented in the SSP.
- Information System Categorization—This is the determination of system's confidentiality, Integrity, and availability requirements.
- Security Classification Guide—For systems that process classified information, this document should detail what data elements are actually classified, their classification, and handling instructions.
- Coding Standards—This artifact informs developers of expectations for producing source code. Details vary from naming conventions to spacing and readability.

Program management tests also include a review of training and maintenance program maturity.

## (2) Education and Training

Education and training material is expected to be developed for various stakeholders tied to development of an information system. This includes managers, developers, and testers. As the focus of program management at this stage is development of a product, training for systems users is not considered in scope.

## (3) Application Maintenance

Application Maintenance addresses processes that support the application through discovery and remediation of flaws.

# **B. DESIGN AND DEVELOPMENT CONSIDERATIONS**

Design and development focuses on the actual creation of an information system. Of the 158 tests in [1], 103 pertain to design and development, by far the largest focus for the guide. Analysis steps in this area focus on assessing how well security was built in throughout creation of the information system and range across many technologies areas.



## (1) Documentation

The first stage of reviewing for design and development considerations includes documentation review. Particular artifacts to assess include:

- Design Document—This artifact addresses the information systems' architecture. It has several security related requirements dealing with interfaces, authentication, and information types.
- Application Configuration Guide—This is the same document identified in program management.
- Threat Model—This artifact represents the efforts made by the developer to identify threats that an information system may encounter and based on the risk each threat represents, determine adequate controls against those threats. It is not uncommon for the threat model to be a component of the design document.

## (2) Best Practices

The next area in the design and development consideration includes various steps and measures that improve the security of an information system but do not fit into any of the other areas discussed later in the chapter. This includes:

- Removal of “dead code.” Dead code is code that has no path for execution.
- Separation of data and presentation services. A common example of meeting this requirement would be installing web servers and database servers on separate hardware or systems.
- Quality assurance measures such as ensuring that directories and file paths are valid.
- Application clean up measures such as deleting temporary files at the end of application sessions.
- Secure default configurations such that when the application is installed or deployed it is already configured to be as secure as possible.
- Comprehensive error handlings so that application failure does not allow the application to enter into an insecure state.

### (3) Cryptography

The cryptography area assesses whether or not approved cryptographic algorithms and technologies are utilized by an information system, where encryption is required. Particular requirements mandate certification and National Security Agency (NSA) approval for classified communications [9].

### (4) Data

The data area assesses how the application stores and transmits data as well as how it connects to its data sources. The ASD STIG [1] requires encryption of data at rest and in transit.

### (5) Authentication

The authentication area assesses how an application performs identification of users and resources. This part of the assessment has checks that vary depending on the technology and authentication requirements for the application. For example, a desktop application that does not utilize networking technologies will have far fewer requirements than a typical web server/database server application.

### (6) Access Control

The access control area assesses how an application grants access to various parts of itself. This part of the assessment reviews access control models and principles, particularly role-based access control and least privilege principles.

### (7) Input Validation

Within design and development considerations, input validation assesses how an application verifies the validity of any data that crosses its trust boundary. This part of the assessment reviews how the application has established criteria for acceptable input and how it defends itself against known input validation exploits. Particular vulnerabilities of interest are:

- SQL Injection Vulnerabilities—Vulnerabilities where an attacker manipulates application queries in order to modify or access data or direct other unauthorized activity of the database or application servers.
- Integer Arithmetic Vulnerabilities—Vulnerabilities where an application generates unpredictable results from integer math. This occurs due to the nature of integers having multiple sizes, signed/unsigned variants, and overflow properties.
- Format String Vulnerabilities—Vulnerabilities where an attacker passes string values specifically formatted for actions for which the application is not designed.
- Command Injection Vulnerabilities—Vulnerabilities where an attacker uses a data injection vulnerability to execute arbitrary commands via the application.
- Cross Site Scripting (XSS) Vulnerabilities—Vulnerabilities where an attacker uploads malicious code to a vulnerable website, allowing for a third party victim to be exploited.
- Cross Site Request Forgery (CSRF) Vulnerabilities—Vulnerabilities where an attacker uses specially crafted HTML to abuse persistent authentication a user may have to another application.
- Buffer Overflow Vulnerabilities—Vulnerabilities where an attacker may write data beyond the allocated memory of the buffer with which he or she is interacting.

#### (8) Canonical Representation

The canonical representation assesses how an application handles the name representation of a resource. This vulnerability is most common for files where the operating systems on which the application resides may have multiple ways to represent such files.

#### (9) Application Information Disclosure

The application information disclosure area assesses the extent to which an application limits disclosure of information that aids an attacker in exploiting the application. Poorly designed applications will disclose valuable information

that facilitates a simpler reconnaissance for an attacker, increasing the likelihood of compromise.

(10) Race Conditions

The race conditions area assesses how well an application manages multiple processes and or threads in an application. Race conditions become most common when multiple processes or threads need access to a single resource.

(11) Auditing

The auditing area assesses how well an application audits its behavior and that of the users and resources that interact with it.

(12) Mobile Code

The mobile code assesses the application's use of mobile code. The Committee on National Security Systems National Information Assurance (IA) Glossary [5] defines mobile code as:

Software programs or parts of programs obtained from remote information systems, transmitted across a network, and executed on a local information system without explicit installation or execution by the recipient.

(13) Internet Protocol Version 6

The Internet Protocol Version 6 (IPV6) area in the design and development considerations assesses whether an application is capable of operating in an IPV6 environment as well as how well it maintains its interoperability with IPV4 environments.

## **C. SOFTWARE CONFIGURATION MANAGEMENT CONSIDERATIONS**

Software Configuration Management (SCM) considerations focus on the creation of an information system. Of the 158 tests in [1], only 4 emphasize SCM. Analysis steps in this area focus on assessing how well configuration of the application was managed over the course of its development life cycle.

### **(1) Software Configuration Management Plan**

The SCM plan area assesses the maturity of the SCM program. The key artifact for this review is the SCM plan. The SCM Plan is expected to identify a myriad of relevant data regarding the application including:

- Roles and responsibilities during development of the application.
- Tools, techniques and methodologies used to develop the application.
- Version and release schedules for the application.

### **(2) Configuration Control Board (CCB).**

The Configuration Control Board (CCB) area assesses the establishment of a CCB.

### **(3) File Integrity**

The file integrity assesses what measures ensure secure transfer of application files. Cryptographic hash technology is used to determine whether or not files have been tampered with.

## **D. TESTING CONSIDERATIONS**

Testing considerations focus on the security-focused testing of an information system. Of the 158 tests in [1], only 11 are specific to testing. Analysis steps in this area focus on identifying security issues before the application is released. For the purpose of this research functional testing and other types of testing are not considered.

#### (1) Test Plans and Procedures

The test plans and procedure area assesses whether or not test procedures have been created and executed before release of the software and also ensures those tests are executed regularly.

#### (2) Fuzz Testing

The fuzz testing area assesses whether the application has had its various data input vectors tested for tolerance to malicious data.

#### (3) Code Coverage

The code coverage area assesses how completely security testing addresses the application as a percentage of total code. Program and Test managers seek to get as close to 100% as possible with low test values indicating a flaw in the test program.

#### (4) Code Reviews

The code review area assesses the processes used to review source code of the application. While code review typically addresses functional issues in the application, in this context security issues are the only focus.

### **E. DEPLOYMENT CONSIDERATIONS**

Deployments considerations focus on security concerns of the application after it has been released and has been deployed to its intended operating environment. Of the 158 tests in [1], 48 emphasize on deployment issues.

#### (1) Documentation

The first stage of reviewing for deployment considerations includes documentation review. The first three documents should have been assessed in the program management considerations review and the last in design and development considerations review. Particular artifacts to assess include:

- System Security Plan (SSP)
- Application Configuration Guide

- Security Classification Guide
- Threat Model

(2) Third-Party Software

The third party software area in the deployment considerations assesses how third party products are configured and hardened when deployed with the application.

(3) Ports and Protocols

The ports and protocols area assesses what ports and protocols an application with network capabilities uses for communication. Both secure use and registration in the DOD Ports, Protocols, and Services database are assessed.

(4) Workplace Security Procedures

The workplace security procedures area assesses traditional security measures implemented to protect physical assets of the application.

(5) Unnecessary Services

The unnecessary services area reviews removal or disabling services and protocols not in use by the application.

(6) Application Maintenance

The application maintenance area assesses the framework to support the application during and after deployment. Here vulnerability and patch management programs are reviewed.

(7) Security Incident Response Process

The security incident response process area assesses the maturity of the incident response process.

(8) Denial of Service

The Denial of Service (DoS) area assesses the applications resilience to DoS attacks. A DoS attack is any malicious activity designed to impede the availability of its target. The assessment includes both resilience of the application and the environment in which it is installed.

(9) Access Control

The access control area assesses the permissions and configuration of application files. In particular, the assessment determines if the application configuration files are configured such that only authorized personnel may modify them.

(10) Database Exports

The database exports area assesses the processes in place to ensure exports of the deployed production database to an application are purged of any sensitive information before being used in development or testing environments.

(11) Public Key Infrastructure Certificate Configuration

The Public Key Infrastructure (PKI) certificate configuration area assesses whether or not the application has been configured to utilize the DOD PKI for authentication and only that infrastructure.

(12) Auditing

The auditing area assesses the depth and breadth of application audit logging as well as how those logs are protected and maintained.

(13) Recovery and Contingency Planning

The recovery and contingency planning area assesses the completeness of the application continuity plans including all relevant recovery procedures.

(14) Account Management



The account management area assesses established processes for account use, creation, modification and deletion. Some of the areas of focus include:

- Password security
- Password complexity
- Shared user accounts
- Privileged access
- Service and Application accounts
- Least privilege policies

(15) Infrastructure Compliance

The infrastructure compliance area in the deployment considerations assesses the documented and confirmed security requirements and compliance with those security requirements for the environment that hosts the application.

(16) Enclave Demilitarized Zone

The enclave Demilitarized Zone (DMZ) area assesses placement of application that communicates across external enclave boundaries in a DMZ.

(17) DOD DMZ

The infrastructure compliance area assesses placement of an Internet facing application within specially demarcated DMZ for outside the Department of Defense Information Networks (DODIN)

## **F. SUMMARY**

In this chapter, the SCA approach used in this research was discussed. The five consideration areas, program management, design and development, software configuration management, testing, and deployment, were enumerated, including the focus areas that compose them. The following chapter will describe the findings that result from applying this SCA approach to MAST.

THIS PAGE INTENTIONALLY LEFT BLANK

## IV. ASSESSMENT FINDINGS

Findings discovered during this research are organized as they are in [1]. As outlined in Appendix A, some findings apply to multiple considerations; those findings will be annotated and discussed in both areas that they apply to.

### A. ASSESSMENT ORGANIZATION

Each STIG Item, represented by a unique SITG ID later in the chapter, is assessed a raw severity category by [1]. Table 1, DISA Severities, lists those severities.

Table 1. DISA Severities, after [1]

Severity		Severity Meaning
High	CAT I	Any vulnerability, the exploitation of which will, directly and immediately result in loss of Confidentiality, Availability, or Integrity.
Moderate	CAT II	Any vulnerability, the exploitation of which has a potential to result in loss of Confidentiality, Availability, or Integrity.
Low	CAT III	Any vulnerability, the existence of which degrades measures to protect against loss of Confidentiality, Availability, or Integrity.

A result from a particular STIG check will be one of the statuses identified in Table 2, STIG Check Status. Typically SCAs focus on findings vice satisfied requirements, so unless there is an item of particular interest, only open findings will be addressed.

Table 2. STIG Check Status

Status	Status Description
Not a Finding	A STIG check is considered “Not a Finding” when the application being assessed meets the requirement in the STIG check.
Not Applicable	A STIG check is considered “Not applicable” when the requirement that the STIG check is assessing is not relevant to the application being assessed.
Open	A STIG check is considered “Open” when the requirement that the STIG check is assessing is not met. Typically the check will clearly indicate what noncompliant behavior or observation drives towards a finding.

## B. OVERVIEW OF FINDINGS

The SCA for MAST culminated in the overall findings described in Table 3, Overall Summary of Findings. With nine (9) open CAT I findings, MAST is considered to have High Risk.

Table 3. Overall Summary of Findings

Severity	Not A Finding	Not Applicable	Open
CAT I	7	17	9
CAT II	15	36	62
CAT III	2	4	6

### C. PROGRAM MANAGEMENT FINDINGS

Table 4, Summary of Program Management Findings, provides a numeric summary of the findings in program management.

Table 4. Summary of Program Management Findings

Severity	Not A Finding	Not Applicable	Open
CAT I	1	0	0
CAT II	2	0	12
CAT III	1	0	0

Table 5, Program Management Findings Details, describes specific findings related to program management. The findings are non-technical in nature. While this may be disarming initially, findings here will generate systemic vulnerability throughout the entire life cycle of the application.

Table 5. Program Management Findings Details, after [1]

STIG ID Severity	Requirement from ASD STIG	Finding Details from SCA
APP2020 Medium	The Program Manager will provide an Application Configuration Guide to the application hosting providers to include a list of all potential hosting enclaves and connection rules and requirements.	Application does not have an Application Configuration Guide.
APP2050 Medium	The Program Manager will ensure the system has been assigned specific Mission Assurance Category (MAC) and confidentiality levels.	System has not been assigned a MAC or confidentiality level.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP2060 Medium	The Program Manager will ensure the development team follows a set of coding standards.	Written record showing implementation of coding standards not available.
APP2110 Medium	The Program Manager and designer will ensure the application is registered with the DOD Ports and Protocols Database.	Application registration in the DOD Ports and Protocols Database has not yet been completed.
APP2120 Medium	The Program Manager will ensure all levels of program management, designers, developers, and testers receive the appropriate security training pertaining to their job function.	No documentation exists ensuring all levels of program management, designers, developers, and testers receive the appropriate security training pertaining to their job function.
APP2130 Medium	The Program Manager will ensure a vulnerability management process is in place to include ensuring a mechanism is in place to notify users, and users are provided with a means of obtaining security updates for the application.	The vulnerability management process is not documented.
APP2140 Medium	The Program Manager will ensure a security incident response process for the application is established that defines reportable incidents and outlines a standard operating procedure for incident response to include Information Operations Condition (INFOCON).	A security incident response process is not documented.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP2150 Medium	The Program Manager will ensure procedures are implemented to assure physical handling and storage of information is in accordance with the data's sensitivity.	Procedures assure physical handling and storage of information is in accordance with the data's sensitivity are not documented.
APP2040 Medium	If the application contains classified data, the Program Manager will ensure a Security Classification Guide exists containing data elements and their classification.	Because MAST could be installed on a classified system, it requires a Security Classification Guide in order to manage any outputs from the tool.
APP2100 Medium	The Program Manager and designer will ensure the application design complies with the DOD Ports and Protocols guidance.	DOD Ports, Protocols, and Services (PPS) Vulnerability Analysis compliance has not yet been determined. Also, new protocol is being deployed to operate the application. The new protocol must be submitted to DOD for risk assessment.
APP2010 Medium	The Program Manager will ensure a System Security Plan (SSP) is established to describe the technical, administrative, and procedural IA program and policies governing the DOD information system, and identifying all IA personnel and specific IA requirements and objectives.	Application does not have a System Security Plan (SSP).

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP2160 Medium	The Program Manager and IAO will ensure development systems, build systems, test systems, and all components comply with all appropriate DOD STIGs, NSA guides, and all applicable DOD policies. The Test Manager will ensure both client and server machines are STIG compliant.	Development is conducted on a non-STIG-compliant platform.

#### **D. DESIGN AND DEVELOPMENT FINDINGS**

Table 6, Summary of Design and Development Findings, provides a numerical summary of the findings in design and development.

Table 6. Summary of Design and Development Findings

Severity	Not A Finding	Not Applicable	Open
CAT I	6	13	8
CAT II	12	27	33
CAT III	1	2	1

Table 7, Design and Development Findings Details, describes specific findings in design and development. The findings are a mix of technical and non-technical.



Table 7. Design and Development Findings Details, after [1]

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP2020 Medium	The Program Manager will provide an Application Configuration Guide to the application hosting providers to include a list of all potential hosting enclaves and connection rules and requirements.	Application does not have an Application Configuration Guide.
APP2060 Medium	The Program Manager will ensure the development team follows a set of coding standards.	Written record showing implementation of coding standards is not available.
APP2100 Medium	The Program Manager and designer will ensure the application design complies with the DOD Ports and Protocols guidance.	DOD Ports, Protocols, and Services (PPS) Vulnerability Analysis compliance has not yet been determined. Also, a new protocol is being deployed to operate the application that has not been registered per instruction [10].
APP2110 Medium	The Program Manager and designer will ensure the application is registered with the DOD Ports and Protocols Database.	Application registration in the DOD Ports and Protocols Database has not yet been completed.
APP3010 Medium	The designer will create and update the Design Document for each release of the application.	Application does not have a Design Document.
APP3020 Medium	The designer will ensure threat models are documented and reviewed for each application release and updated as required by design and functionality changes or new threats are discovered.	A threat model was not developed for the application.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP3100 Medium	The Designer will ensure the application removes temporary storage of files and cookies when the application is terminated.	Application does not store temporary authentication data, however server does not clean up logs, which may contain sensitive information due to the mission and capability of this software.
APP3150 Medium	The designer will ensure the application uses the Federal Information Processing Standard (FIPS) 140-2 validated cryptographic modules and random number generator if the application implements encryption, key exchange, digital signature, and hash functionality.	Encryption is not used by the application.
APP3170 Medium	The designer will ensure the application uses encryption to implement key exchange and authenticate endpoints prior to establishing a communication channel for key exchange.	Encryption is not used by the application.
APP3210 Medium	The designer will ensure the appropriate cryptography is used to protect stored DOD information if required by the information owner.	Encryption is not used by the application.
APP3220 Medium	The designer will ensure sensitive data held in memory is cryptographically protected when not in use, if required by the information owner, and classified data held in memory is always cryptographically protected when not in use.	Encryption is not used by the application.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP3230 Medium	The designer will ensure the application properly clears or overwrites all memory blocks used to process sensitive data, if required by the information owner, and clears or overwrites all memory blocks used for classified data.	Because of the nature of MAST, it is possible to load sensitive data into memory. The application does not have the ability to clear or overwrite memory blocks used to process sensitive data.
APP3240 Medium	The designer will ensure all access authorizations to data are revoked prior to initial assignment, allocation or reallocation to an unused state.	Design document does not exist.
APP3250 High	The designer will ensure data transmitted through a commercial or wireless network is protected using an appropriate form of cryptography.	Encryption is not used by the application.
APP3260 Medium	The designer will ensure the application uses mechanisms assuring the integrity of all transmitted information (including labels and security parameters).	Application does not use any kind of integrity mechanism.
APP3270 High	The designer will ensure the application has the capability to mark sensitive/classified output when required.	Application does not have the ability to mark sensitive or classified data. Documentation indicating how to handle sensitive or classified data is also absent.
APP3300 Medium	The designer will ensure applications requiring server authentication are PK-enabled.	Application does not have server authentication mechanism. Lack of capability is a finding. A PKI waiver would be required to continue without being PKI enabled.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP3450 Medium	The designer and IAO will ensure application resources are protected with permission sets which allow only an application administrator to modify application resource configuration files.	As a desktop application, the ability to ensure application resources are protected with any permission sets is not possible.
APP3470 Medium	The designer will ensure the application is organized by functionality and roles to support the assignment of specific roles to specific application functions.	Application does not implement a roles capability to organize functionality.
APP3480 High	The designer will ensure access control mechanisms exist to ensure data is accessed and changed only by authorized personnel.	Application does not deploy any kind of access control, leaving its features, capabilities and functions unprotected.
APP3500 Medium	The designer will ensure the application executes with no more privileges than necessary for proper operation.	The nature of this application requires that it have significantly more permissions than the average desktop application.
APP3510 High	The designer will ensure the application validates all input.	Application test plans are not available, indicating not all input validation vulnerabilities are tested for, as required by ASD STIG. Additionally, code review sample revealed that input is not validated in multiple methods including: ClientProgram.main(), NetworkLoader(),NetworkLoader.readJSONFromFile()
APP3550 High	The designer will ensure the application is not vulnerable to integer arithmetic issues.	Application has numerous areas in source code that do not check for integer overflows. Particular potential issues are in the scenario parsing files, where integers read from a file without validation.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP3560 High	The designer will ensure the application does not contain format string vulnerabilities.	Potential format string vulnerabilities exist in <code>mastSEServer.ServerCLI.print(String target)</code> . Application does not appear to validate input,
APP3570 High	The designer will ensure the application does not allow command injection.	Application test plans are not available, indicating command injection vulnerabilities may not be tested for, as required by ASD STIG. Additionally, code review sample revealed that the <code>ClientProgram.main()</code> method lacks sufficient input validation in network input calls to inhibit a command injection attack.
APP3590 High	The designer will ensure the application does not have buffer overflows, use functions known to be vulnerable to buffer overflows, and does not use signed values for memory allocation where permitted by the programming language.	Application test plans are not available, indicating buffer overflow vulnerabilities may not be tested for, as required by ASD STIG. Additionally, code review sample revealed that the <code>ScenarioValidator.validate()</code> method reads multiple strings from a file and does not validate buffer sizes allowing for potential overflows.
APP3600 Medium	The designer will ensure the application has no canonical representation vulnerabilities.	Application test plans are not available, indicating canonical representation vulnerabilities may not be tested for, as required by ASD STIG. Additionally, code review sample revealed that the <code>SEServer</code> class has methods that consume file paths without protective measures including paths to log and scenario files.
APP3620 Medium	The designer will ensure the application does not disclose unnecessary information to users.	Application has exceptions that present stack traces to users at runtime.



<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP3630 Medium	The designer will ensure the application is not vulnerable to race conditions.	Application test plans are not available, indicating race conditions vulnerabilities are not tested for.
APP3650 Low	The designer will ensure the application has a capability to notify an administrator when audit logs are nearing capacity as specified in the system documentation.	Application does notify an administrator when audit logs are nearing capacity and no specification exists in system documentation.
APP3670 Medium	The designer will ensure the application has a capability to display the user's time and date of the last change in data content.	Application does not have the capability to display the user's time and date of the last change in data content.
APP3680 Medium	The designer will ensure the application design includes audits on all access to need-to-know information and key application events.	Application does not audit all required fields.
APP3690 Medium	The designer and IAO will ensure the audit trail is readable only by the application and auditors and protected against modification and deletion by unauthorized individuals.	Application audit trail is not protected from modification and deletion by unauthorized individuals.
APP3700 Medium	The designer will ensure unsigned Category 1A mobile code is not used in the application in accordance with DOD policy.	Application's Simware modules contain various forms of mobile code, which are unsigned.
APP3710 Medium	The designer will ensure signed Category 1A and Category 2 mobile code signature is validated before executing.	Application performs no validation of mobile code.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Finding Details from SCA</b>
APP3720 Medium	The designer will ensure unsigned Category 2 mobile code executing in a constrained environment has no access to local system and network resources.	Category 2 mobile code may be required in the Simware this application deploys, requiring that this control not be enforced.
APP3730 Medium	The designer will ensure uncategorized or emerging mobile code is not used in applications.	Emerging mobile code can be implemented depending on the Simware deployed by the application. Emerging code cannot be used and should be submitted for risk assessment. [11]
APP3750 Medium	The designer will ensure development of new mobile code includes measures to mitigate the risks identified.	Risk Mitigation plans have not been developed for any mobile code that is deployed. [11]
APP3960 Medium	The designer will ensure the application is compliant with all DOD IT Standards Registry (DISR) IPv6 profiles.	Application is not IPv6 capable.
APP3970 Medium	The designer will ensure supporting application services and interfaces have been designed, or upgraded for, IPv6 transport.	Application is not IPv6 capable.
APP3980 Medium	The designer will ensure the application is compliant with IPv6 multicast addressing and features an IPv6 network configuration options as defined in RFC 4038.	Application is not IPv6 capable.
APP3990 Medium	The designer will ensure the application is compliant with the IPv6 addressing scheme as defined in RFC 1884.	Application is not IPv6 capable.

## **E. SOFTWARE CONFIGURATION MANAGEMENT FINDINGS**

Table 8, Summary of Software Configuration Management Findings, provides a numeric summary of the findings in software configuration management.

Table 8. Summary of Software Configuration Management Findings

Severity	Not A Finding	Not Applicable	Open
CAT I	0	0	0
CAT II	0	0	3
CAT III	0	0	1

Table 9, Software Configuration Management Findings Details, describes specific findings in software configuration management. The findings are non-technical in nature.



Table 9. Software Configuration Management Findings Details, after  
[1]

<b>STIG ID Severity</b>	<b>Requirement</b>	<b>Finding Details</b>
APP4010 Low	The Release Manager will ensure the access privileges to the configuration management (CM) repository are reviewed every 3 months.	A Software Configuration Management program has not been developed for the application.
APP4030 Medium	The Release Manager will develop an SCM plan describing the configuration control and change management process of objects developed and the roles and responsibilities of the organization.	A Software Configuration Management program has not been developed for the application.
APP404 Medium0	The Release Manager will establish a Configuration Control Board (CCB), which meets at least every release cycle, for managing the CM process.	A Software Configuration Management program has not been developed for the application.
APP4050 Medium	The release manager must ensure application files are cryptographically hashed prior to deploying to DOD operational networks.	A Software Configuration Management program has not been developed for the application.

## F. TESTING FINDINGS

Table 10, Summary of Testing Findings, provides a numerical summary of the findings in testing.

Table 10. Summary of Testing Findings

Severity	Not A Finding	Not Applicable	Open
CAT I	0	0	0
CAT II	1	0	8
CAT III	0	0	2

Table 11, Testing Findings Details, describes specific findings testing. The findings are non-technical in nature.

Table 11. Testing Findings Details, after [1]

STIG ID Severity	Requirement	Finding Details
APP2160 Medium	The Program Manager and IAO will ensure development systems, build systems, test systems, and all components comply with all appropriate DOD STIGs, NSA guides, and all applicable DOD policies. The Test Manager will ensure both client and server machines are STIG compliant.	Development on a STIG-compliant platform is not documented.
APP5010 Low	The Test Manager will ensure at least one tester is designated to test for security flaws in addition to functional testing.	No one is designated to test the application for security flaws

<b>STIG ID Severity</b>	<b>Requirement</b>	<b>Finding Details</b>
APP5040 Medium	The Test Manager will ensure the changes to the application are assessed for IA and accreditation impact prior to implementation.	Changes to the application are not assessed for IA and accreditation impact prior to implementation.
APP5050 Medium	The Test Manager will ensure tests plans and procedures are created and executed prior to each release of the application or updates to system patches.	Test plans, procedures, and results do not exist.
APP5060 Medium	The Test Manager will ensure test procedures are created and at least annually executed to ensure system initialization, shutdown, and aborts are configured to ensure the system remains in a secure state.	Test plans, procedures, and results do not exist.
APP5070 Low	The Test Manager will ensure code coverage statistics are maintained for each release of the application.	Code coverage statistics are not maintained.
APP5080 Medium	The Test Manager will ensure a code review is performed before the application is released.	Application code is not being reviewed.
APP5090 Medium	The Test Manager will ensure flaws found during a code review are tracked in a defect tracking system.	Application code is not being reviewed.
APP5100 Medium	The IAO will ensure active vulnerability testing is performed.	Vulnerability testing is not performed.
APP5110 Medium	The Test Manager will ensure security flaws are fixed or addressed in the project plan.	The application does not have a project plan.

## G. DEPLOYMENT FINDINGS

Table 12, Summary of Deployment Findings, provides a numerical summary of the findings in deployment.

Table 12. Summary of Deployment Findings

<b>Severity</b>	<b>Not A Finding</b>	<b>Not Applicable</b>	<b>Open</b>
CAT I	0	4	0
CAT II	1	14	23
CAT III	1	2	2

Table 13, Deployment Findings Details, describes the specific findings of deployment. These findings typically apply to applications later in their life cycle than MAST; however, many still do apply.

Table 13. Deployment Findings Details, after [1]

<b>STIG ID Severity</b>	<b>Requirement</b>	<b>Finding Details</b>
APP2010 Medium	The Program Manager will ensure a System Security Plan (SSP) is established to describe the technical, administrative, and procedural IA program and policies governing the DOD information system, and identifying all IA personnel and specific IA requirements and objectives.	Application does not have a system security plan (SSP).
APP2020 Medium	The Program Manager will provide an Application Configuration Guide to the application hosting providers to include a list of all potential hosting enclaves and connection rules and requirements.	Application does not have an Application Configuration Guide.
APP2040 Medium	If the application contains classified data, the Program Manager will ensure a Security Classification Guide exists containing data elements and their classification.	Because MAST could be installed on a classified system, requires Security Classification Guide in order to manage any outputs from the tool.
APP2100 Medium	The Program Manager and designer will ensure the application design complies with the DOD Ports and Protocols guidance.	DOD Ports, Protocols, and Services (PPS) Vulnerability Analysis compliance has not yet been determined. Also, a new protocol is being deployed to operate the application and must be submitted to DOD for risk assessment. [10].
APP2110 Medium	The Program Manager and designer will ensure the application is registered with the DOD Ports and Protocols Database.	Application registration in the DOD Ports and Protocols Database has not yet been completed.

<b>STIG ID Severity</b>	<b>Requirement</b>	<b>Finding Details</b>
APP2140 Medium	The Program Manager will ensure a security incident response process for the application is established that defines reportable incidents and outlines a standard operating procedure for incident response to include Information Operations Condition (INFOCON).	A security incident response process is not documented.
APP2150 Medium	The Program Manager will ensure procedures are implemented to assure physical handling and storage of information is in accordance with the data's sensitivity.	Procedures assure physical handling and storage of information is in accordance with the data's sensitivity are not documented.
APP2160 Medium	The Program Manager and IAO will ensure development systems, build systems, test systems, and all components comply with all appropriate DOD STIGs, NSA guides, and all applicable DOD policies. The Test Manager will ensure both client and server machines are STIG compliant.	Development on a STIG compliant platform is not documented.
APP3020 Medium	The designer will ensure threat models are documented and reviewed for each application release and updated as required by design and functionality changes or new threats are discovered.	A threat model was not developed for the application.

<b>STIG ID Severity</b>	<b>Requirement</b>	<b>Finding Details</b>
APP3450 Medium	The designer and IAO will ensure application resources are protected with permission sets which allow only an application administrator to modify application resource configuration files.	As a desktop application, the ability to ensure application resources are protected with any permission sets is not possible.
APP3470 Medium	The designer will ensure the application is organized by functionality and roles to support the assignment of specific roles to specific application functions.	Application does not implement a roles capability.
APP3690 Medium	The designer and IAO will ensure the audit trail is readable only by the application and auditors and protected against modification and deletion by unauthorized individuals.	Application audit trail is not protected from modification and deletion by unauthorized individuals.
APP6010 Medium	The IAO will ensure if an application is designated critical, the application is not hosted on a general purpose machine.	The nature of MAST requires that it be co-hosted with whatever applications reside on the platform that it will operate on. This risk must be accepted.
APP6030 Medium	The IAO will ensure unnecessary services are disabled or removed.	No unnecessary services are enabled for the application.
APP6040 Medium	The IAO will ensure at least one application administrator has registered to receive update notifications, or security alerts, when automated alerts are available.	Update notifications are not available for the application.



STIG ID Severity	Requirement	Finding Details
APP6050 Medium	The IAO will ensure the system and installed applications have current patches, security updates, and configuration settings.	A configuration management plan, including procedures, does not exist.
APP6060 High	The IAO will ensure the application is decommissioned when maintenance or support is no longer available.	Application is not yet under maintenance.
APP6080 Medium	The IAO will ensure protections against DoS attacks are implemented.	Application does not have a threat model.
APP6110 Low	The IAO will review audit trails periodically based on system documentation recommendations or immediately upon system security events.	Application does not audit all required fields to make periodic review an effective control.
APP6130 Low	The IAO will ensure, for classified systems, application audit trails are continuously and automatically monitored, and alerts are provided immediately when unusual or inappropriate activity is detected.	System's confidentiality level has not yet been assigned, which prevents completion of this assessment. Because the possibility exists that it may operate in a classified environment, this must remain an open finding.
APP6140 Medium	The IAO will ensure application audit trails are retained for at least 1 year for applications without SAMI data, and 5 years for applications including SAMI data.	Application does not maintain log entries for intervals of that length.



<b>STIG ID Severity</b>	<b>Requirement</b>	<b>Finding Details</b>
APP6160 Medium	The IAO will ensure recovery procedures and technical system features exist so recovery is performed in a secure and verifiable manner. The IAO will document circumstances inhibiting a trusted recovery.	Application does not have a disaster recovery plan.
APP6170 Medium	The IAO will ensure back-up copies of the application software are stored in a fire-rated container and not collocated with operational software.	No documentation on data backups was available.
APP6180 Medium	The IAO will ensure procedures are in place to assure the appropriate physical and technical protection of the backup and restoration of the application.	No documentation on data backups was available.
APP6190 Medium	The IAO will ensure data backup is performed at required intervals in accordance with DOD policy.	No documentation on data backups was available. Application likely does not require backups, however there is no documentation to state so.
APP6200 Medium	The IAO will ensure a disaster recovery plan exists in accordance with DOD policy based on the Mission Assurance Category (MAC).	Application does not have a disaster recovery plan and also does not have a Mission Assurance Category (MAC) assigned to drive that plan's creation.

## **H. SUMMARY**

In this chapter, the results of the SCA of MAST were enumerated. As documented in Table 3. Overall Summary of Findings, MAST had 9 CAT I, 62 CAT II, and 6 CAT III findings. Typically CAT III findings are overlooked in DoN SCA practice. The CAT I and CAT II findings, however, require mitigation efforts. Chapter V documents a threat model approach to the address the most dangerous vulnerabilities in MAST.

## **V. MITIGATION AND REMEDIATION THROUGH THREAT MODELING**

This chapter describes architecture changes and vulnerability mitigations to improve MAST's overall security posture.

In order to deploy effective vulnerability mitigations to an application its threat model must be understood. A threat model is a representation of threats to an application, prioritization those threats, and application of countermeasures and mitigations. A threat model must be conducted early in an application development life cycle.

For MAST, or any DOD system, some threat modeling burden is relieved via the application and implementation of DISA STIGs. The STIGs are designed to address vulnerabilities commonly found in "the wild." For example, according to [12], the top application vulnerability is injection flaws such as SQL injections attacks for web applications. To address this common weakness, [1] has multiple checks for input validation and injection attack defense. Despite this "pre-completed" modeling.

For the purpose of this research a threat model is created and analyzed under the context of improving MAST's security posture. Unlike typical threat models which are intended to occur as part of the design process, this threat model will be used to funnel security posture improvement effort into major areas of concern. Again, as the research is conducted outside of MAST's development process, some tailoring liberty is required to model after the fact; such tailoring will be explained as needed. Also, the intent is not to include detailed tutorial on threat modeling but rather utilize existing techniques, as such general detail on how to build a threat model should be sought through other resources.

## **A. THREAT MODEL BASICS**

According to [13], there are nine steps to developing a security model:

1. Define use scenarios
2. List external dependencies
3. Define security assumptions
4. Create external security notes
5. Create data flow diagrams (DFD)
6. Determine threat types
7. Identify threats to the system
8. Determine risk
9. Plan mitigations

The following sections establish the threat model for MAST.

## **B. MAST USE SCENARIOS**

A Use Scenario describes how the subject of a threat model is used.

As discussed in Chapter II, MAST is a desktop application designed to simulate malware on operational platforms. It consists of three desktop application components in a three-tier architecture in which modules called Simware and Scenario files are passed between the tiers to execute simulations.

## **C. MAST EXTERNAL DEPENDENCIES**

External dependencies in the context of threat models are other pieces of software and hardware that are not within the scope of the software being modeled but may have an impact on what threats it will face.

As MAST is designed to be deployed to operational platforms its external dependencies vary wildly. Some predictable expectations include:

- Installation on a Microsoft Windows operating system
- Dependency on the Java virtual machine to run
- Networking support

#### **D. MAST SECURITY ASSUMPTIONS**

Security assumptions are prerequisites and expectations the software being modeled may have for its dependencies and environment, particularly in the support of security. Examples might include encryption services and file permissions implemented by a hosting operating system.

In the case of MAST, no security assumptions can be made based on the documentation available at the time of this research.

#### **E. MAST DATA FLOW DIAGRAMS**

DFDs demonstrate how the application interacts with its own components and external interfaces.

Several DFDs are required to describe MAST. DFD elements are numbered to facilitate discussion.

Figure 2, MAST DFD with MAST Components, shows how the various components of MAST interact for command and control (C2). This data flow demonstrates how the various tiers of MAST communicate. To meet the needs of this exercise, only “Request” and “Response” transaction need be modeled. In this DFD, a request is an instruction from one tier of MAST to a lower tier. The response represents some acknowledgement from the lower tier to the higher tier.

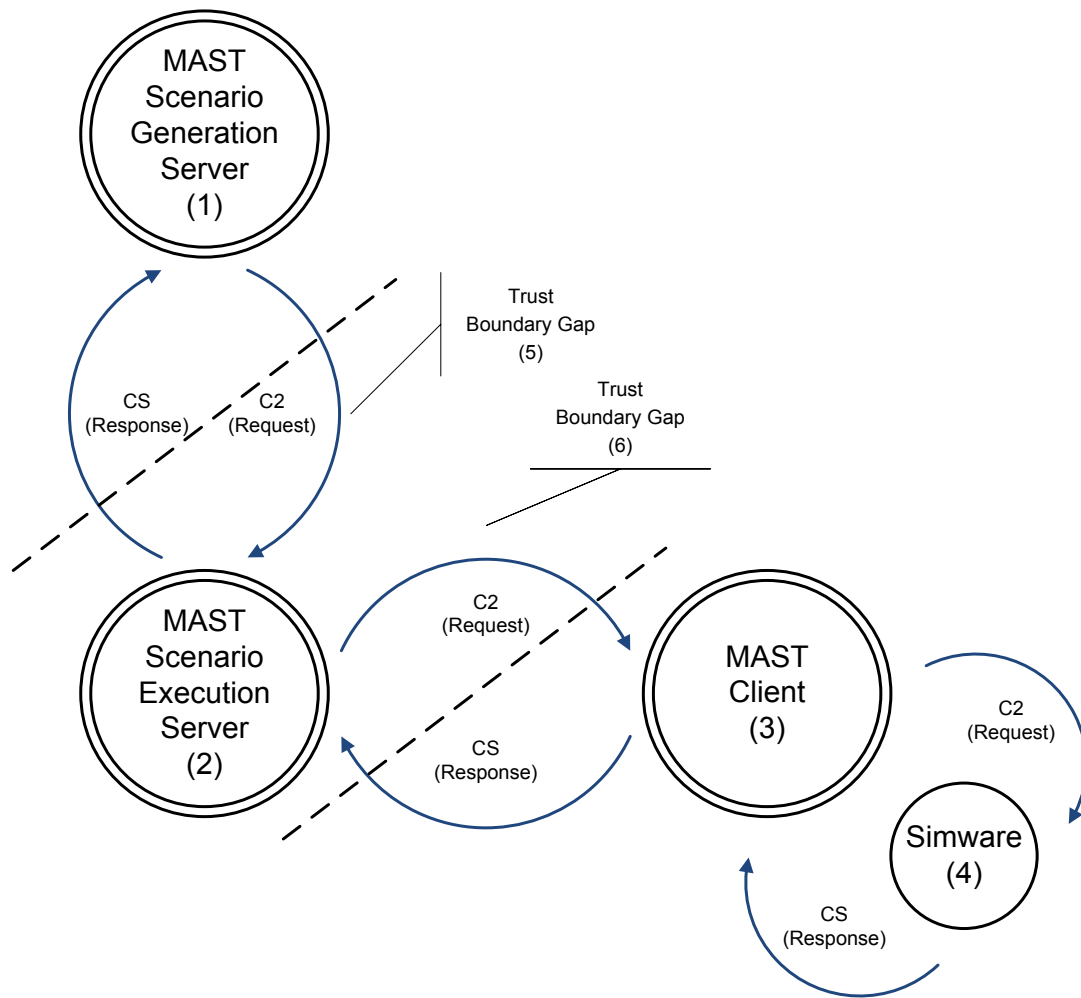


Figure 2. MAST DFD with MAST Components

Figure 3, MAST Simware and Scenario File Distribution, shows how MAST Simware and Scenario files are passed from the MAST SG Server to the MAST Scenario ES through to the MAST Client.

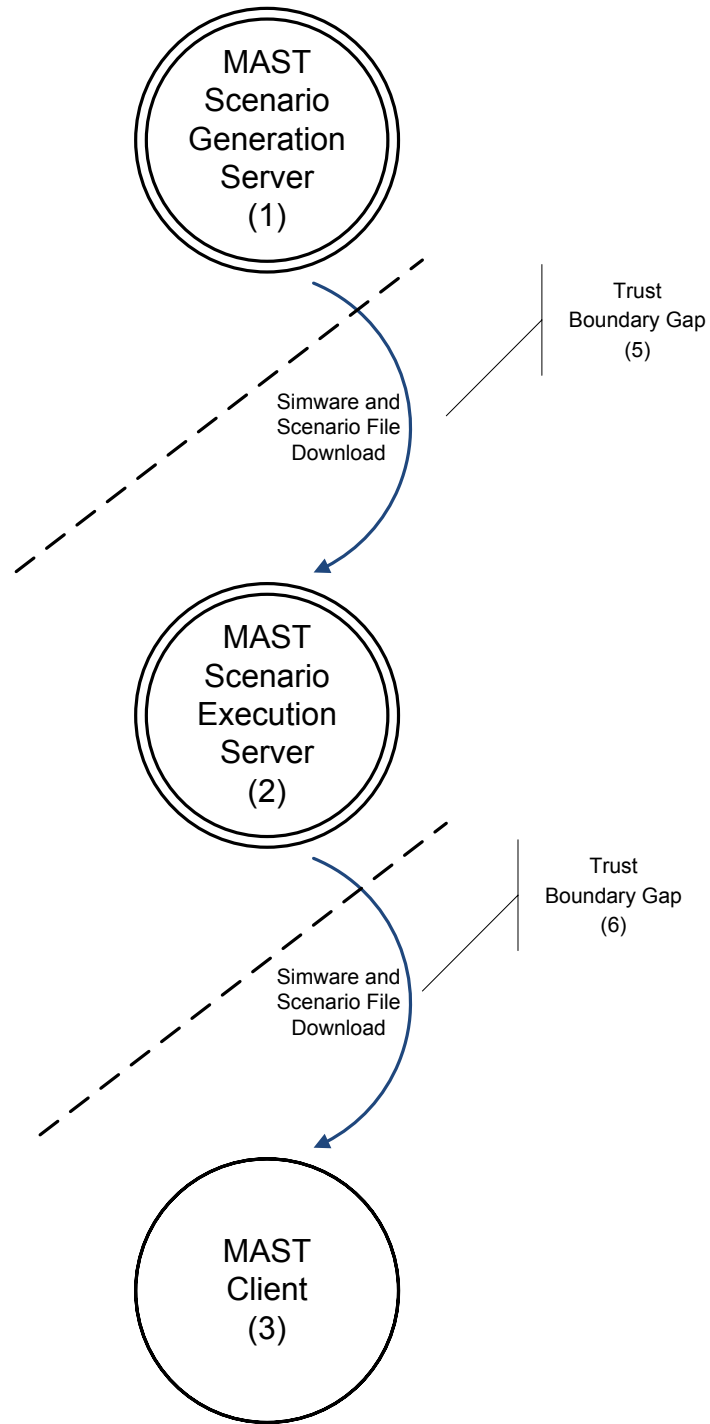


Figure 3. MAST Simware and Scenario File Distribution

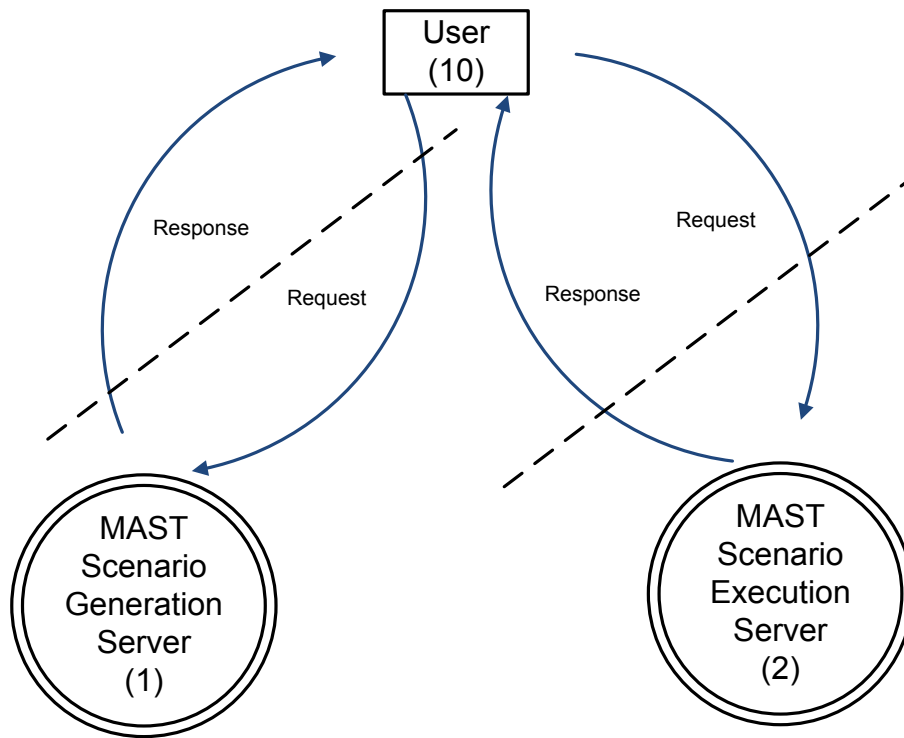


Figure 4. MAST DFD with User

Figure 4, MAST DFD with User, shows user interaction with the MAST SG Server and MAST SE Server however due to MAST being a desktop application; this particular DFD will not be used. It is only provided for completion.

Each DFD's items are numbered to facilitate discussion throughout the threat model. When an item appears in multiple models its number assignment is consistent.

## F. MAST THREAT TYPES

This stage of threat modeling determines how to categorize threat types. This categorization is crucial for the next step of the model. The simplest approach utilizes the threats to confidentiality, integrity, and availability where threats are categorized by their impact to those classic security pillars. This is the approach that is used in this research. Threat type definitions are as follows:



- Confidentiality Threats—Threats that allow for information disclosure and access to unauthorized parties
- Integrity Threats—Threats that allow for modification of data by unauthorized entities or in an unauthorized manner
- Availability Threats—Threats that affect the accessibility of authorized entities

## G. IDENTIFY THREATS TO MAST

Threat identification consists of mapping of the established threat types to elements or data flows between elements. Table 14, DFD Item Inventory, lists each item in the DFDs. The “Boundary Gap” DFD Elements Type is a unique type used for the purpose of MAST threat model. It acts as a place holder for whenever data flowed outside the application boundary. Its significance will be discussed later in the paper.

Table 14. DFD Item Inventory

DFD Elements Types	DFD Item Numbers
External Entities	User (10)
Processes	SG Server (1) SE Server (2) MAST Client (3) Simware (4)
Data Flows	SG to SE Request (1→2) SE to SG Response (2→1) SE to Client Request (2→3) Client to SE Response (3→2) Client to Simware Request (3→4) Simware to Client Response (4→3) SG to SE Simware/Scenario Download (1→2) SE to Client Simware/Scenario Download (2→3) User to SG Request (10→1) SG to User Response (1→10) User to SE Request (10→2) SE to User Response (2→10)
Boundary GAP	Trust Boundary Gap (5) Trust Boundary Gap (6)

In order to simplify and reduce the redundancy and complexity of the model, similar data paths can be collapsed as a single dataflow. Table 15, Reduced DFD Item Inventory, shows an inventory where common data flows are simplified.

Table 15. Reduced DFD Item Inventory

DFD Elements Types	DFD Item Numbers
External Entities	User (10)
Processes	SG Server (1) SE Server (2) MAST Client (3) Simware (4)
Data Flows	SG to SE C2 (1→2) SE to Client C2 (2→3) Client to Simware C2 (3→4) SG to SE Simware/Scenario Download (1→2) SE to Client Simware/Scenario Download (2→3)
Boundary GAP	Trust Boundary Gap (5) Trust Boundary Gap (6)

Table 16, Threats to MAST, shows a simplified listing of the threats that MAST must defend against.

Table 16. Threats to MAST

Threat Type	Effected DFD Item
Confidentiality	SG to SE C2 and Downloads (1→2) SE to Client C2 and Downloads (2→3)
Integrity	SG to SE C2 and Downloads (1→2) SE to Client C2 and Downloads (2→3) Simware (4)
Availability	N/a

## **H. DETERMINE RISK TO MAST**

Once threats have been established, the risk those threats pose to the subject system must be determined. This helps prioritize resources to remediate and mitigate vulnerabilities.

There are many methodologies available to assess risk of a threat. To keep aligned with this research's DOD SCA focus, severities from [1] are used to assess the risk of vulnerabilities.

The Confidentiality threat risk is established in the following controls from [1] where the SCA generated findings:

- APP3250—High Risk
- APP3330—High Risk
- APP3340—High Risk
- APP3405—High Risk
- APP3210—Medium Risk
- APP3260—Medium Risk
- APP3150—Medium Risk
- APP3170—Medium Risk
- APP3220—Medium Risk
- APP3900—Medium Risk
- APP3950—Medium Risk
- APP2070—Low Risk

In accordance with DOD risk management practice the highest risk rating of any particular control becomes the risk rating of any group controls of which it is a part. In this risk model, Confidentiality related controls were grouped together. Because several controls are high risk per [1], the overall risk of the group is high risk

The Integrity threat risk is established in the following controls from [1] where the SCA generated findings:

- APP3700—Medium Risk
- APP3710—Medium Risk
- APP3720—Medium Risk
- APP3730—Medium Risk
- APP3740—Medium Risk
- APP3750—Medium Risk

These STIG severities require that these risks be assessed as medium risk. As in the Confidentiality section, Integrity related controls were grouped together in the risk model, and the highest risk assessed to any of the controls is medium. Thus, the overall risk of the Integrity group is medium

Due to the sensitive nature of MASTs purpose, simulating malware on operational platforms, thereby reducing the risk beyond the severity in [1] without taking technical measure to actually closing the finding is inappropriate.

## **I. PLAN MITIGATIONS FOR MAST**

Once risks have been prioritized and measured, mitigation should follow. Risk response typically has several different approaches and these approaches can be mixed and blended to serve the purpose of the program or organization. As defined in [14], these approaches are:

- Risk Acceptance—This risk response approach occurs when a program or organization tolerates the risk as assessed and takes no measures to change the risk as it is assessed.
- Risk Avoidance—This risk response approach is the result of a program or organization halting the activity or endeavor that causes the risk to be generated in the first place.
- Risk Mitigation—This risk response approach is due to the program or organization taking measures to reduce the risk or otherwise alter the risk as it is assessed.

- Risk Sharing or Transfer—This risk response approach is selected when the program or organization uses a third party to absorb some or all of the risk as assessed. Typically this strategy is related to insurance.

Changes to system architecture and other technical improvements to an IS are considered risk mitigations. Thus, for the purpose of this research, only mitigations to risk will be discussed.

#### (1) Application Tier-to-Tier Trust

MAST is designed as a three-tiered application with each tier having a network stack and capability of being placed on separate hosts. This separation injects “trust boundaries” outside of which MAST cannot guarantee the confidentiality and integrity of the communications between its components.

In order to meet the confidentiality and integrity needs, encryption needs to be deployed. MAST uses the java.net library for communication between its three tiers; specifically the Socket and ServerSocket classes are used. This is, of course, the unencrypted socket library. The Java language has an SSL library, javax.net.ssl, with SSLSocket and SSLServerSocket classes [15]. These libraries can very rapidly provide for the encryption and in turn confidentiality and integrity needs of MAST.

In addition to encryption, an authentication scheme must be deployed to ensure clients and servers within MAST communicate with only the entities that are intended. There is no simple solution here. MAST would need to undergo a significant change to support an authentication scheme. As MAST is designed to run on an operational platform with built in communication capabilities, it may be designed to utilize a symmetric encryption and inherit public key infrastructure (PKI) capabilities for key and password distribution.

#### (2) Simware Trust

Simware are the core of the MAST architecture. These modules perform the actual simulation of malicious activity. MAST Clients execute Simware with

no validation of source or capabilities of module. There is little to stop MAST from being a botnet vice a learning tool due to the absence of these validation capabilities. Code signing and validation technologies could be used to mitigate this finding.

## **J. THREAT MODEL LIMITATIONS**

Threat models will only directly model threats of a technical nature. Issues related to Program Management, Configuration Management, and early design decisions and related rigor will not be apparent unless subjected to a SCA or similar audit. Many of the findings discovered in the SCA are attributable to nontechnical issues, thus mitigations in this area are relatively simple as MAST is so early in its development life cycle.

Table 17, Mitigations Not Found in Threat Model, provides overview of findings that would not be apparent in a threat model and also proposes mitigations to those findings.

Table 17. Mitigations Not Found in Threat Model, after [1]

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP2010 Medium	The Program Manager will ensure a System Security Plan (SSP) is established to describe the technical, administrative, and procedural IA program and policies governing the DOD information system, and identifying all IA personnel and specific IA requirements and objectives.	Develop a System Security Plan (SSP) that addresses implementation of all IA controls and supporting policies and processes for MAST.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP2020 Medium	The Program Manager will provide an Application Configuration Guide to the application hosting providers to include a list of all potential hosting enclaves and connection rules and requirements.	Develop a Application Configuration Guide that describes how to deploy MAST to a target system or host.
APP2040 Medium	If the application contains classified data, the Program Manager will ensure a Security Classification Guide exists containing data elements and their classification.	Develop a Security Classification Guide describing data elements within MAST and their classifications if any.
APP2050 Medium	The Program Manager will ensure the system has been assigned specific MAC and confidentiality levels.	Assign a MAC and confidentiality levels for MAST.
APP2060 Medium	The Program Manager will ensure the development team follows a set of coding standards.	Establish coding standards for the MAST development team
APP2100 Medium	The Program Manager and designer will ensure the application design complies with the DOD Ports and Protocols guidance.	Design MAST to comply with DOD Ports and Protocols guidance. This will include proper registration of MAST protocols DOD Ports and Protocols Database.
APP2110 Medium	The Program Manager and designer will ensure the application is registered with the DOD Ports and Protocols Database.	Design MAST to comply with DOD Ports and Protocols guidance. This will include proper registration of MAST protocols DOD Ports and Protocols Database.
APP2120 Medium	The Program Manager will ensure all levels of program management, designers, developers, and testers receive the appropriate security training pertaining to their job function.	Ensure all personnel supporting MAST receive at a minimum security awareness training. Additional training might include secure software development and secure coding curriculum

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP2130 Medium	The Program Manager will ensure a vulnerability management process is in place to include ensuring a mechanism is in place to notify users, and users are provided with a means of obtaining security updates for the application.	Develop a vulnerability management process such that MAST users are notified of updates and have a means go acquiring updates.
APP2140 Medium	The Program Manager will ensure a security incident response process for the application is established that defines reportable incidents and outlines a standard operating procedure for incident response to include Information Operations Condition (INFOCON).	Develop an incident response process for MAST that outlines and to resolve and report incidents that affect the application.
APP2150 Medium	The Program Manager will ensure procedures are implemented to assure physical handling and storage of information is in accordance with the data's sensitivity.	Develop procedures for secure handling of MAST physical media.
APP2160 Medium	The Program Manager and IAO will ensure development systems, build systems, test systems, and all components comply with all appropriate DOD STIGs, NSA guides, and all applicable DOD policies. The Test Manager will ensure both client and server machines are STIG compliant.	Configure MAST development and testing environments to meet and comply with all appropriate DOD STIGs, NSA guides, and all applicable DOD policies.
APP3010 Medium	The designer will create and update the Design Document for each release of the application.	Develop and execute processes that keep application design documentation update with each release.



<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP3020 Medium	The designer will ensure threat models are documented and reviewed for each application release and updated as required by design and functionality changes or new threats are discovered.	Develop threat models to properly represent the threats that face MAST. The model built for this research is a starting point for this requirement.
APP3100 Medium	The Designer will ensure the application removes temporary storage of files and cookies when the application is terminated.	Design MAST to remove any temporary files that it may create. This would include removal of Simware files that are downloaded to clients during routine MAST use.
APP3230 Medium	The designer will ensure the application properly clears or overwrites all memory blocks used to process sensitive data, if required by the information owner, and clears or overwrites all memory blocks used for classified data.	Design MAST to overwrite any memory it utilizes. Because MAST has the potential to gleam sensitive information as it runs its simulations, it must clean up and reference to that information within its own boundary.
APP3240 Medium	The designer will ensure all access authorizations to data are revoked prior to initial assignment, allocation or reallocation to an unused state.	MAST has no authorization or authentication process, making compliance with this requirement impossible. Ensure design of a subsequent authorization or authentication process revokes access authorizations.
APP3270 High	The designer will ensure the application has the capability to mark sensitive/classified output when required.	MAST has no Security Classification Guide and no indicator for what confidentiality it will operate at, thus it cannot meet the capability to mark sensitive/classified output when required. Satisfy the Security Classification Guide and confidentiality assignment to meet this requirement.
APP3300 Medium	The designer will ensure applications requiring server authentication are PK-enabled.	MAST is likely not a likely candidate for PKI enablement and as such, a PKI waiver should be pursued

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP3450 Medium	The designer and IAO will ensure application resources are protected with permission sets which allow only an application administrator to modify application resource configuration files.	MAST, as a desktop application, likely does not need to provide any additional protection to its resources beyond what is already provided by its host. Including configuration for this in the Application Configuration Guide would resolve this finding.
APP3470 Medium	The designer will ensure the application is organized by functionality and roles to support the assignment of specific roles to specific application functions.	MAST, as a desktop application, likely does not need to support this functionality. Including configuration for roles in the application configuration guide would resolve this finding.
APP3480 High	The designer will ensure access control mechanisms exist to ensure data is accessed and changed only by authorized personnel.	MAST, as a desktop application, likely does not need to support this functionality. Including configuration for access control mechanisms in the application configuration guide would resolve this finding.
APP3500 Medium	The designer will ensure the application executes with no more privileges than necessary for proper operation.	MAST's application configuration guide must document that it executes with no more privileges than necessary for proper operation. Actual installation must support this as well. Because MAST will emulate and simulate threatening behavior by design, it will need advanced privileged access on the hosts it is deployed. This must be documented and understood clearly.
APP3510 High	The designer will ensure the application validates all input.	All MAST input fields must validate input. Code review processes can find these issues and also demonstrate their resolution. Test plans must be developed, executed, and include tests for this kind of vulnerability.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP3550 High	The designer will ensure the application is not vulnerable to integer arithmetic issues.	All MAST input fields must validate input. Code review processes can find these issues and also demonstrate their resolution. Test plans must be developed, executed, and include tests for this kind of vulnerability.
APP3560 High	The designer will ensure the application does not contain format string vulnerabilities.	All MAST input fields must validate input. Code review processes can find these issues and also demonstrate their resolution. Test plans must be developed, executed, and include tests for this kind of vulnerability.
APP3570 High	The designer will ensure the application does not allow command injection.	All MAST input field must validate input. Code review processes can find these issues and also demonstrate their resolution. Test plans must be developed, executed, and include tests for this kind of vulnerability.
APP3590 High	The designer will ensure the application does not have buffer overflows, use functions known to be vulnerable to buffer overflows, and does not use signed values for memory allocation where permitted by the programming language.	All MAST input fields must validate input. Code review processes can find these issues and also demonstrate their resolution. Test plans must be developed, executed, and include tests for this kind of vulnerability.
APP3600 Medium	The designer will ensure the application has no canonical representation vulnerabilities.	All MAST input fields must validate input. Code review processes can find these issues and also demonstrate their resolution. Test plans must be developed, executed, and include tests for this kind of vulnerability.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP3620 Medium	The designer will ensure the application does not disclose unnecessary information to users.	MAST will present users with stack traces on some errors. Code review processes can find these issues and also demonstrate their resolution. Corrections include updating try-catch blocks to address errors more cleanly and ensuring all potential errors are caught.
APP3630 Medium	The designer will ensure the application is not vulnerable to race conditions.	Code review processes find these issues and also demonstrate their resolution. Test plans must be developed, executed, and include tests for this kind of vulnerability.
APP3670 Medium	The designer will ensure the application has a capability to display the user's time and date of the last change in data content.	Develop robust logging for MAST. With the current minimal logging capabilities, MAST cannot display the user's time and date of the last change in data content.
APP3680 Medium	The designer will ensure the application design includes audits on all access to need-to-know information and key application events.	Develop robust logging for MAST. Mast has only minimal logging capabilities and cannot support audits on access to need-to-know information and events.
APP3690 Medium	The designer and IAO will ensure the audit trail is readable only by the application and auditors and protected against modification and deletion by unauthorized individuals.	Develop robust logging for MAST. Mast has only minimal logging capabilities.
APP3960 Medium	The designer will ensure the application is compliant with all DOD IT Standards Registry (DISR) IPv6 profiles.	Java has IPv6 support built in and will attempt to handle IPv6/IPv4 issues seamlessly, however application test plans need to be created and executed to ensure the compatibility.
APP3970 Medium	The designer will ensure supporting application services and interfaces have been designed, or upgraded for, IPv6 transport.	Java has IPv6 support built in and will attempt to handle IPv6/IPv4 issues seamlessly, however application test plans need to be created and executed to ensure the compatibility.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP3980 Medium	The designer will ensure the application is compliant with IPv6 multicast addressing and features an IPv6 network configuration options as defined in RFC 4038.	Java has IPv6 support built in and will attempt to handle IPv6/IPv4 issues seamlessly, however application test plans need to be created and executed to ensure the compatibility.
APP3990 Medium	The designer will ensure the application is compliant with the IPv6 addressing scheme as defined in RFC 1884.	Java has IPv6 support built in and will attempt to handle IPv6/IPv4 issues seamlessly, however application test plans need to be created and executed to ensure the compatibility.
APP4030 Medium	The Release Manager will develop an SCM plan describing the configuration control and change management process of objects developed and the roles and responsibilities of the organization.	Develop a SCM plan for MAST. It must describe configuration control and change management process
APP4040 Medium	The Release Manager will establish a Configuration Control Board (CCB) that meets at least every release cycle, for managing the CM process.	Establish a CCB to support and execute the SCM for MAST.
APP4050 Medium	The release manager must ensure application files are cryptographically hashed prior to deploying to DOD operational networks.	Within the SCM, create processes to hash new releases of MAST. Also, ensure that those hashes are included in MAST releases.
APP5040 Medium	The Test Manager will ensure the changes to the application are assessed for IA and accreditation impact prior to implementation.	MAST does not have a documented test program. In developing that program, include the development and execution IA tests for MAST.
APP5050 Medium	The Test Manager will ensure tests plans and procedures are created and executed prior to each release of the application or updates to system patches.	Develop a test program for MAST. Ensure, as part of the program, that tests prior to release.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP5060 Medium	The Test Manager will ensure test procedures are created and at least annually executed to ensure system initialization, shutdown, and aborts are configured to ensure the system remains in a secure state.	Develop a test program for MAST. Ensure, as part of the program, that tests are included that the system remains in a secure state, through shutdown, startup, and other overall system state changes.
APP5080 Medium	The Test Manager will ensure a code review is performed before the application is released.	Develop a test program for MAST. Ensure, as part of the program, that code review is included.
APP5090 Medium	The Test Manager will ensure flaws found during a code review are tracked in a defect tracking system.	Develop a test program for MAST. Ensure, as part of the program, that findings from code review are tracked and resolved.
APP5100 Medium	The IAO will ensure active vulnerability testing is performed.	Develop and execute a vulnerability management program.
APP5110 Medium	The Test Manager will ensure security flaws are fixed or addressed in the project plan.	Develop a test program for MAST. Ensure, as part of the program, that security flaws are fixed.
APP6010 Medium	The IAO will ensure if an application is designated critical, the application is not hosted on a general purpose machine.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.
APP6030 Medium	The IAO will ensure unnecessary services are disabled or removed.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.
APP6040 Medium	The IAO will ensure at least one application administrator has registered to receive update notifications, or security alerts, when automated alerts are available.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP6050 Medium	The IAO will ensure the system and installed applications have current patches, security updates, and configuration settings.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.
APP6060 High	The IAO will ensure the application is decommissioned when maintenance or support is no longer available.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.
APP6080 Medium	The IAO will ensure protections against DoS attacks are implemented.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.
APP6140 Medium	The IAO will ensure application audit trails are retained for at least 1 year for applications without SAMI data, and 5 years for applications including SAMI data.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.
APP6160 Medium	The IAO will ensure recovery procedures and technical system features exist so recovery is performed in a secure and verifiable manner. The IAO will document circumstances inhibiting a trusted recovery.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.
APP6170 Medium	The IAO will ensure back-up copies of the application software are stored in a fire-rated container and not colocated with operational software.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.
APP6180 Medium	The IAO will ensure procedures are in place to assure the appropriate physical and technical protection of the backup and restoration of the application.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.

<b>STIG ID Severity</b>	<b>Requirement from ASD STIG</b>	<b>Mitigation</b>
APP6190 Medium	The IAO will ensure data backup is performed at required intervals in accordance with DOD policy.	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.
APP6200 Medium	The IAO will ensure a disaster recovery plan exists in accordance with DOD policy based on the Mission Assurance Category (MAC).	MAST development platforms and environments require an IAO and IA processes to ensure secure operation. Establish IAO to corrected findings.

## **K. SUMMARY**

In this chapter, a threat model for MAST was developed and then utilized to propose mitigations to vulnerabilities in MAST's architecture. Particular areas of concern were determined to be in Application Tier-to-Tier Trust and Simware Trust. Following the threat model, findings that would not be apparent in a threat model are enumerated along with their corresponding mitigations. The next chapter concludes the research and discusses future research opportunities.



## **VI. CONCLUSION AND FUTURE WORK**

This chapter discusses conclusions from this research and opportunities for future work.

### **A. CONCLUSION**

This thesis was driven by the need to determine the weaknesses and vulnerabilities within MAST and steps to be taken to prepare it for deployment on an operational platform, focusing specifically on security concerns.

The SCA was executed based on [1], the primary guidance document for DOD secure application development and deployment. Following the SCA, a risk analysis was conducted via threat modeling methodology. The threat model defines vulnerabilities to confidentiality, availability, and integrity and supported mitigations approaches to resolve those findings.

In Chapter IV, the SCA determined that MAST has several CAT I, CAT II and CATIII findings. Should these findings be left unaddressed, MAST would not be able to acquire an ATO.

In Chapter V, the threat model determined critical risk areas in which future efforts should be targeted to get MAST ready for deployment.

At this stage in its development, MAST is not ready for deployment in an operation environment, beyond strictly limited and controlled test events. This research has shown that MAST has several major rectifications necessary to operate securely.

Once properly hardened, MAST will provide unique opportunity for advancing information system operators' cyber readiness. MAST shows great promise in its utilization as a training device to ready the cyber workforce to manage real-life cyber risks.

## **B. FUTURE WORK**

The SCA has identified several opportunities for improvement to MAST, following what has been done in this thesis, particularly to implement vulnerability mitigations. Each of the vulnerabilities identified by this research is significant enough to warrant future research and development.

### **1. Application Tier-to-Tier Trust**

The Tier-to-Tier Trust problem requires significant research and development in order to determine how to effectively meet confidentiality and integrity needs. Key-exchange for encryption between tiers and authentication represent significant challenges that warrant further research.

### **2. Simware Trust**

In order to deploy a complete code signing and validation program MAST may require significant extensions. Success of a secure distribution system would require a single source for Simware in order to protect signing signatures. Because of the intensity of change to MAST to support this, there is great opportunity in pursuing this researching and developing this solution.

### **3. Additional SCA**

This SCA was based on version 3, release 8 of the ASD STIG [1], which was the current release when this research began. The ASD STIG has been updated twice while this research was underway and upon completion, the ASD STIG was at version 3, release 10. As secure development practices are constantly evolving and improving, these updates are to be expected. If the mitigations proposed in this research are implemented, a very different and much more secure MAST would result, creating the opportunity to research another SCA.

## APPENDIX. DISA APPLICATION SECURITY STIG AREAS

Table 18. DISA Application Security STIG Areas, shows what focus areas each test from the Application Security and Development STIG apply to. An “X” indicates that a test applies to that particular focus area.

Table 18. DISA Application Security STIG Areas, after [1]

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
V-6127	The designer will ensure applications requiring user authentication are PK-enabled and are designed and implemented to support hardware tokens (e.g., CAC for NIPRNet).	IATS-1, IATS-2		X			
V-6128	The designer and IAO will ensure PK-enabled applications are designed and implemented to use approved credentials authorized under the DOD PKI program.	IATS-1, IATS-2		X			
V-6129	The designer will ensure the application using PKI validates certificates for expiration, confirms origin is from a DOD authorized CA, and verifies the certificate has not been revoked by CRL or OCSP, and CRL cache (if used) is updated at least daily.	IATS-1, IATS-2		X			
V-6130	The designer will ensure the application has the capability to require account passwords that conform to DOD policy.	IAIA-1		X			X
V-6131	The designer will ensure the application prevents the creation of duplicate accounts.	IAIA-1		X			
V-6132	The IAO will ensure all user accounts are	IAAC-1, IAIA-1					X

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	disabled which are authorized to have access to the application but have not authenticated within the past 35 days.						
V-6133	The IAO will ensure unnecessary built-in application accounts are disabled.	IAIA-1					X
V-6134	The IAO will ensure default passwords are changed.	IAIA-1					X
V-6135	The designer will ensure the appropriate cryptography is used to protect stored DOD information if required by the information owner.	ECCR-1, ECCR-2, ECCR-3		X			
V-6136	The designer will ensure data transmitted through a commercial or wireless network is protected using an appropriate form of cryptography.	ECCT-1, ECCT-2, ECNK-1, ECNK-2		X			
V-6137	The designer will ensure the application uses the Federal Information Processing Standard (FIPS) 140–2 validated cryptographic modules and random number generator if the application implements encryption, key exchange, digital signature, and hash functionality.	DCNR-1, ECCR-1, ECCR-2, ECCT-1, ECCT-2		X			
V-6138	The designer will ensure the application design includes audits on all access to need-to-know information and key application events.	ECAR-1, ECAR-2, ECAR-3		X			
V-6139	The designer will ensure the application has a capability to notify an administrator when audit logs are nearing capacity as specified in the system documentation.	ECAT-2		X			
V-6140	The designer and IAO will ensure the audit trail is readable only by the	ECTP-1		X			X

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	application and auditors and protected against modification and deletion by unauthorized individuals.						
V-6141	The designer will ensure access control mechanisms exist to ensure data is accessed and changed only by authorized personnel.	ECDD-2, ECLP-1, ECPA-1		X			
V-6142	The designer will ensure all access authorizations to data are revoked prior to initial assignment, allocation or reallocation to an unused state.	ECRC-1		X			
V-6143	The designer will ensure the application executes with no more privileges than necessary for proper operation.	ECLP-1		X			
V-6144	The designer will ensure the application provides a capability to limit the number of logon sessions per user and per application.	ECLO-1		X			
V-6145	If the application contains classified data, the Program Manager will ensure a Security Classification Guide exists containing data elements and their classification.	DCSD-1	X				X
V-6146	The designer will ensure the application has the capability to mark sensitive/classified output when required.	ECML-1		X			
V-6147	The Test Manager will ensure the application does not modify data files outside the scope of the application.	ECRC-1				X	
V-6148	The designer will ensure threat models are documented and reviewed for each application release and updated as	DCSQ-1		X			X

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	required by design and functionality changes or new threats are discovered.						
V-6149	The designer will ensure the application does not contain source code that is never invoked during operation, except for software components and libraries from approved third-party products.	DCSQ-1		X			
V-6150	The Designer will ensure the application does not store configuration and control files in the same directory as user data.	DCPA-1		X			
V-6151	The IAO will ensure unnecessary services are disabled or removed.	DCSD-1					X
V-6152	The designer will ensure the application is capable of displaying a customizable click-through banner at logon which prevents further activity on the information system unless and until the user executes a positive action to manifest agreement by clicking on a box indicating 'OK.'	ECWM-1		X			
V-6153	The designer will ensure the application removes authentication credentials on client computers after a session terminates.	IAIA-1, IAIA-2		X			
V-6154	The designer will ensure the application is organized by functionality and roles to support the assignment of specific roles to specific application functions.	ECLP-1, ECPA-1		X			X
V-6155	The designer will ensure the application provides a capability to terminate a session and log out.	DCSQ-1		X			
V-6156	The designer will ensure the application	IAIA-1, IAIA-2		X			

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	does not contain embedded authentication data.						
V-6157	The designer will ensure the application does not contain invalid URL or path references.	DCSQ-1		X			
V-6158	The designer will ensure the application only embeds mobile code in email which does not execute automatically when the user opens the email body or attachment.	DCMC-1		X			
V-6159	The designer will ensure unsigned Category 1A mobile code is not used in the application in accordance with DOD policy.	DCMC-1		X			
V-6160	The designer will ensure unsigned Category 2 mobile code executing in a constrained environment has no access to local system and network resources.	DCMC-1		X			
V-6161	The designer will ensure signed Category 1A and Category 2 mobile code signature is validated before executing.	DCMC-1		X			
V-6162	The designer will ensure uncategorized or emerging mobile code is not used in applications.	DCMC-1		X			
V-6163	The Designer will ensure the application removes temporary storage of files and cookies when the application is terminated.	ECRC-1		X			
V-6164	The designer will ensure the application validates all input.	DCSQ-1		X			
V-6165	The designer will ensure the application does not have buffer overflows, use functions known to be vulnerable to buffer	DCSQ-1		X			

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	overflows, and does not use signed values for memory allocation where permitted by the programming language.						
V-6166	The designer will ensure the application is not subject to error handling vulnerabilities.	DCSQ-1		X			
V-6167	The designer will ensure application initialization, shutdown, and aborts are designed to keep the application in a secure state.	DCSS-2		X			
V-6168	The designer will ensure applications requiring server authentication are PK-enabled.	IATS-1, IATS-2		X			
V-6169	The Program Manager and designer will ensure the application design complies with the DOD Ports and Protocols guidance.	DCPP-1	X	X			X
V-6170	The Program Manager and designer will ensure any IA, or IA enabled, products used by the application are NIAP approved or in the NIAP approval process.	DCAS-1	X	X			
V-6171	The IAO will ensure recovery procedures and technical system features exist so recovery is performed in a secure and verifiable manner. The IAO will document circumstances inhibiting a trusted recovery.	CODP-1, CODP-2, CODP-3					X
V-6172	The IAO will ensure data backup is performed at required intervals in accordance with DOD policy.	CODB-1, CODB-2, CODB-3					X
V-6173	The IAO will ensure application audit	ECRR-1					X



Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	trails are retained for at least 1 year for applications without SAMI data, and 5 years for applications including SAMI data.						
V-6174	The IAO will ensure production database exports have database administration credentials and sensitive data removed before releasing the export.	ECAN-1					X
V-6197	The Program Manager will ensure a System Security Plan (SSP) is established to describe the technical, administrative, and procedural IA program and policies governing the DOD information system, and identifying all IA personnel and specific IA requirements and objectives.	DCSD-1	X				X
V-6198	The Program Manager and IAO will ensure development systems, build systems, test systems, and all components comply with all appropriate DOD STIGs, NSA guides, and all applicable DOD policies. The Test Manager will ensure both client and server machines are STIG compliant.	DCCS-1, DCCS-2, ECSC-1	X			X	X
V-7013	The designer will create and update the Design Document for each release of the application.	DCFA-1		X			
V-16773	The Program Manager will provide an Application Configuration Guide to the application hosting providers to include a list of all potential hosting enclaves and connection rules and requirements.	DCID-1, EBCR-1	X	X			X
V-16775	The Program Manager will ensure the	DCSD-1	X				

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	system has been assigned specific MAC and confidentiality levels.						
V-16776	The Program Manager will ensure the development team follows a set of coding standards.	DCSQ-1	X	X			
V-16777	The Program Manager will ensure COTS IA and IA enabled products, comply with NIAP/NSA endorsed protection profiles.	DCSR-1, DCSR-2, DCSR-3	X				
V-16778	The Program Manager will document and obtain DAA risk acceptance for all public domain, shareware, freeware, and other software products/libraries with both (1) no source code to review, repair, and extend, and (2) limited or no warranty, when such products are required for mission accomplishment.	DCPD-1	X	X			
V-16779	The Program Manager and designer will ensure the application is registered with the DOD Ports and Protocols Database.	DCPP-1	X	X			X
V-16780	The Program Manager will ensure all levels of program management, designers, developers, and testers receive the appropriate security training pertaining to their job function.	PRTN-1	X				
V-16781	The Program Manager will ensure a vulnerability management process is in place to include ensuring a mechanism is in place to notify users, and users are provided with a means of obtaining security updates for the application.	DCCT-1, VIVM-1	X				
V-16782	The Program Manager will ensure a security incident response process for the	VIIR-1, VIIR-2	X				X

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	application is established that defines reportable incidents and outlines a standard operating procedure for incident response to include Information Operations Condition (INFOCON).						
V-16783	The Program Manager will ensure procedures are implemented to assure physical handling and storage of information is in accordance with the data's sensitivity.	PESP-1	X				X
V-16784	The designer will ensure the user interface services are physically or logically separated from data storage and management services.	DCPA-1		X			
V-16785	The designer will ensure the application supports detection and/or prevention of communication session hijacking.	ECTM-2		X			
V-16786	The designer will ensure the application installs with unnecessary functionality disabled by default.	DCSD-1		X			
V-16787	The designer will ensure the application follows the secure failure design principle.	DCSQ-1		X			
V-16788	The designer will ensure the application uses encryption to implement key exchange and authenticate endpoints prior to establishing a communication channel for key exchange.	DCNR-1		X			
V-16789	The designer will ensure private keys are accessible only to administrative users.	ECCD-1		X			
V-16790	The designer will ensure the application does not connect to a database using administrative credentials or other	ECLP-1		X			

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	privileged database accounts.						
V-16791	The designer will ensure transaction based applications implement transaction rollback and transaction journaling.	ECDC-1		X			
V-16792	The designer will ensure sensitive data held in memory is cryptographically protected when not in use, if required by the information owner, and classified data held in memory is always cryptographically protected when not in use.	ECCR-1, ECCR-2, ECCR-3		X			
V-16793	The designer will ensure the application properly clears or overwrites all memory blocks used to process sensitive data, if required by the information owner, and clears or overwrites all memory blocks used for classified data.	ECCR-1, ECCR-2, ECCR-3		X			
V-16794	The designer will ensure the application uses mechanisms assuring the integrity of all transmitted information (including labels and security parameters).	ECTM-1, ECTM-2		X			
V-16795	The designer will ensure the application does not display account passwords as clear text.	IAIA-1		X			
V-16796	The designer will ensure the application transmits account passwords in an approved encrypted format.	ECCT-1		X			
V-16797	The designer will ensure the application stores account passwords in an approved encrypted format.	IAIA-1, IAIA-2		X			
V-16798	The designer will ensure the application	ECCD-1		X			

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	protects access to authentication data by restricting access to authorized users and services.						
V-16799	The designer will ensure the application installs with unnecessary accounts disabled, or deleted, by default.	IAIA-1		X			
V-16800	The designer will ensure users' accounts are locked after three consecutive unsuccessful logon attempts within one hour.	ECLO-1, ECLO-2		X			
V-16801	The designer will ensure locked users' accounts can only be unlocked by the application administrator.	ECLO-1		X			
V-16802	The designer will ensure the application provides a capability to automatically terminate a session and log out after a system defined session idle time limit is exceeded.	ECLO-1		X			
V-16803	The designer and IAO will ensure application resources are protected with permission sets which allow only an application administrator to modify application resource configuration files.	ECCD-1		X			X
V-16804	The designer will ensure the application does not rely solely on a resource name to control access to a resource.	DCSQ-1		X			
V-16806	The designer will ensure the web application assigns the character set on all web pages.	DCSQ-1		X			
V-16807	The designer will ensure the application is not vulnerable to SQL Injection, uses prepared or parameterized statements,	DCSQ-1, ECCD-1		X			

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	does not use concatenation or replacement to build SQL queries, and does not directly access the tables in a database.						
V-16808	The designer will ensure the application is not vulnerable to integer arithmetic issues.	DCSQ-1		X			
V-16809	The designer will ensure the application does not contain format string vulnerabilities.	DCSQ-1		X			
V-16810	The designer will ensure the application does not allow command injection.	DCSQ-1		X			
V-16811	The designer will ensure the application does not have cross site scripting (XSS) vulnerabilities.	DCSQ-1		X			
V-16812	The designer will ensure the application has no canonical representation vulnerabilities.	DCSQ-1		X			
V-16813	The designer will ensure the application does not use hidden fields to control user access privileges or as a part of a security mechanism.	DCSQ-1		X			
V-16814	The designer will ensure the application does not disclose unnecessary information to users.	ECCD-1		X			
V-16815	The designer will ensure the application is not vulnerable to race conditions.	DCSQ-1		X			
V-16816	The designer will ensure the application supports the creation of transaction logs for access and changes to the data.	ECCD-2		X			
V-16817	The designer will ensure the application	ECLO-2		X			

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	has a capability to notify the user of important login information.						
V-16818	The designer will ensure the application has a capability to display the user's time and date of the last change in data content.	ECCD-2		X			
V-16819	The designer will ensure development of new mobile code includes measures to mitigate the risks identified.	DCMC-1		X			
V-16820	The Release Manager will ensure the access privileges to the configuration management (CM) repository are reviewed every 3 months.	ECPC-1, ECPC-2			X		
V-16822	The Release Manager will develop an SCM plan describing the configuration control and change management process of objects developed and the roles and responsibilities of the organization.	DCPR-1, DCSW-1			X		
V-16823	The Release Manager will establish a Configuration Control Board (CCB), that meets at least every release cycle, for managing the CM process.	DCCB-1, DCCB-2			X		
V-16824	The Test Manager will ensure at least one tester is designated to test for security flaws in addition to functional testing.	DCSQ-1				X	
V-16825	The Test Manager will ensure the changes to the application are assessed for IA and accreditation impact prior to implementation.	DCII-1				X	
V-16826	The Test Manager will ensure tests plans and procedures are created and executed	DCCT-1				X	

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	prior to each release of the application or updates to system patches.						
V-16827	The Test Manager will ensure test procedures are created and at least annually executed to ensure system initialization, shutdown, and aborts are configured to ensure the system remains in a secure state.	DCSS-2				X	
V-16828	The Test Manager will ensure code coverage statistics are maintained for each release of the application.	DCSQ-1				X	
V-16829	The Test Manager will ensure a code review is performed before the application is released.	DCSQ-1				X	
V-16830	The Test Manager will ensure flaws found during a code review are tracked in a defect tracking system.	DCSQ-1				X	
V-16831	The IAO will ensure active vulnerability testing is performed.	DCSQ-1				X	
V-16832	The Test Manager will ensure security flaws are fixed or addressed in the project plan.	DCSQ-1				X	
V-16833	The IAO will ensure if an application is designated critical, the application is not hosted on a general purpose machine.	DCSQ-1					X
V-16834	The IAO shall ensure if a DOD STIG or NSA guide is not available, a third-party product will be configured by the following in descending order as available: 1) commercially accepted practices, (2) independent testing results, or (3) vendor literature.	DCCS-1					X



Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
V-16835	The IAO will ensure at least one application administrator has registered to receive update notifications, or security alerts, when automated alerts are available.	DCCT-1					X
V-16836	The IAO will ensure the system and installed applications have current patches, security updates, and configuration settings.	DCCT-1					X
V-16837	The IAO will ensure the application is decommissioned when maintenance or support is no longer available.	DCSD-1, ECSC-1					X
V-16838	Procedures are not in place to notify users when an application is decommissioned.	DCSD-1					X
V-16839	The IAO will ensure protections against DoS attacks are implemented.	DCSQ-1					X
V-16840	The IAO will ensure the system alerts an administrator when low resource conditions are encountered.	ECAT-2					X
V-16841	The IAO will review audit trails periodically based on system documentation recommendations or immediately upon system security events.	ECCD-2					X
V-16842	The IAO will report all suspected violations of IA policies in accordance with DOD information system IA procedures.	ECAT-2					X
V-16843	The IAO will ensure, for classified systems, application audit trails are continuously and automatically monitored, and alerts are provided	ECAT-2					X

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	immediately when unusual or inappropriate activity is detected.						
V-16844	The IAO will ensure back-up copies of the application software are stored in a fire-rated container and not collocated with operational software.	COSW-1					X
V-16845	The IAO will ensure procedures are in place to assure the appropriate physical and technical protection of the backup and restoration of the application.	COBR-1					X
V-16846	The IAO will ensure a disaster recovery plan exists in accordance with DOD policy based on the Mission Assurance Category (MAC).	CODB-1, CODB-2, CODP-3					X
V-16847	The IAO will ensure an account management process is implemented, verifying only authorized users can gain access to the application, and individual accounts designated as inactive, suspended, or terminated are promptly removed.	IAAC-1					X
V-16848	The IAO will ensure passwords generated for users are not predictable and comply with the organization's password policy.	IAIA-1, IAIA-2					X
V-16849	The IAO will ensure the application's users do not use shared accounts.	IAGA-1					X
V-16850	The IAO will ensure connections between the DOD enclave and the Internet or other public or commercial wide area networks require a DMZ.	EBPW-1					X
V-19687	The IAO will ensure web servers are on	DCPA-1					X

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	logically separate network segments from the application and database servers if it is a tiered application.						
V-19688	The designer and the IAO will ensure physical operating system separation and physical application separation is employed between servers of different data types in the web tier of Increment 1/Phase 1 deployment of the DOD DMZ for Internet-facing applications.	DCPA-1					X
V-19689	The designer will ensure web services are designed and implemented to recognize and react to the attack patterns associated with application-level DoS attacks.	DCSQ-1		X			
V-19690	The designer will ensure the web service design includes redundancy of critical functions.	DCSQ-1		X			
V-19691	The designer will ensure web service design of critical functions is implemented using different algorithms to prevent similar attacks from forming a complete application level DoS.	DCSQ-1		X			
V-19692	The designer will ensure web services are designed to prioritize requests to increase availability of the system.	DCSQ-1		X			
V-19693	The designer will ensure execution flow diagrams are created and used to mitigate deadlock and recursion issues.	DCSQ-1		X			
V-19694	The IAO will ensure an XML firewall is deployed to protect web services.	DCSQ-1					X
V-19695	The designer will ensure web services	DCSQ-1		X			

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	provide a mechanism for detecting resubmitted SOAP messages.						
V-19696	The designer and IAO will ensure digital signatures exist on UDDI registry entries to verify the publisher.	DCSQ-1		X			X
V-19697	The designer and IAO will ensure UDDI versions are used supporting digital signatures of registry entries.	DCSQ-1		X			X
V-19698	The designer and IAO will ensure UDDI publishing is restricted to authenticated users.	DCSQ-1		X			X
V-19699	The IAO will ensure web service inquiries to UDDI provide read-only access to the registry to anonymous users.	ECLP-1					X
V-19700	The IAO will ensure if the UDDI registry contains sensitive information and read access to the UDDI registry is granted only to authenticated users.	ECCR-1, ECCR-2					X
V-19701	The designer will ensure SOAP messages requiring integrity, sign the following message elements:-Message ID-Service Request-Timestamp-SAML Assertion (optionally included in messages)	ECTM-1		X			
V-19702	The designer will ensure when using WS-Security, messages use timestamps with creation and expiration times.	ECTM-2, IAIA-2		X			
V-19703	The designer will ensure validity periods are verified on all messages using WS-Security or SAML assertions.	IAIA-2		X			
V-19704	The designer shall ensure each unique asserting party provides unique assertion	IAIA-2		X			

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	ID references for each SAML assertion.						
V-19705	The designer shall ensure encrypted assertions, or equivalent confidentiality protections, when assertion data is passed through an intermediary, and confidentiality of the assertion data is required to pass through the intermediary.	IAIA-2		X			
V-19706	The designer will ensure the application is compliant with all DOD IT Standards Registry (DISR) IPv6 profiles.	DCSQ-1		X			
V-19707	The designer will ensure supporting application services and interfaces have been designed, or upgraded for, IPv6 transport.	DCSQ-1		X			
V-19708	The designer will ensure the application is compliant with IPv6 multicast addressing and features an IPv6 network configuration options as defined in RFC 4038.	DCSQ-1		X			
V-19709	The designer will ensure the application is compliant with the IPv6 addressing scheme as defined in RFC 1884.	DCSQ-1		X			
V-21498	The designer will ensure the application is not vulnerable to XML Injection.	DCSQ-1		X			
V-21500	The designer will ensure the application does not have CSRF vulnerabilities.	DCSQ-1		X			
V-21519	The Program Manager will ensure all products are supported by the vendor or the development team.	DCSQ-1	X				
V-22028	The designer shall use the NotOnOrAfter property when using the <SubjectConfirmation> element in a	DCSQ-1		X			

Vuln ID	Rule Title	IA Controls	Program Management	Design and Development	Software Configuration Management	Testing	Deployment
	SAML assertion.						
V-22029	The designer shall use both the <NotBefore> and <NotOnOrAfter> elements or <OneTimeUse> element when using the <Conditions> element in a SAML assertion.	DCSQ-1		X			
V-22030	The designer will ensure the asserting party uses FIPS approved random numbers in the generation of SessionIndex in the SAML element AuthnStatement.	DCSQ-1		X			
V-22031	The designer shall ensure messages are encrypted when the SessionIndex is tied to privacy data.	ECNK-1		X			
V-22032	The designer shall ensure if a OneTimeUse element is used in an assertion, there is only one used in the Conditions element portion of an assertion.	DCSQ-1		X			
V-47163	The release manager must ensure application files are cryptographically hashed prior to deploying to DOD operational networks.	DCSQ-1			X		

## LIST OF REFERENCES

- [1] Defense Information Systems Agency, "Application security and development: Security technical implementation guide" Department of Defense, Washington, DC, STIG V3R8, July 2014
- [2] L. Ray, Jr., "Scalability assessments for the Malicious Activity Simulation Tool (MAST)," M.S. thesis, Dept. Comput. Sci., Naval Postgraduate School, Monterey, CA, 2013.
- [3] N. J. Hayes, "A definitive interoperability test methodology for the Malicious Activity Simulation Tool (Mast)," M.S. thesis, Dept. Comput. Sci., Naval Postgraduate School, Monterey, CA, 2013.
- [4] A. M. Littlejohn and E. Makhoulf, "Test and evaluation of the Malicious Activity Simulation Tool (MAST) in a Local Area Network (LAN) running the Common PC Operating System Environment (COMPOSE)." M.S. thesis, Dept. Comput. Sci., Naval Postgraduate School, Monterey, CA, 2013.
- [5] *National Information Assurance Glossary*, CNSS 4009, Committee on National Security Systems, Department of Defense, Washington, DC, 2010.
- [6] National Institute of Standards and Technology, "Managing information security risk: Organization, mission, and information system view," NIST, Gaithersburg, MD, Special Publication 800–39, Mar. 2011.
- [7] National Institute of Standards and Technology, "Guide for applying the risk management framework to federal information systems," Special Publication 800–37 Revision 1, Feb. 2011.
- [8] *Risk Management Framework (RMF) for DOD Information Technology (IT)*, DOD Instruction 8510.01, Department of Defense, Washington, DC, 2014.
- [9] Security requirements for cryptographic modules, Federal Information Processing Standards Publication 140–2, 1994
- [10] *Ports, Protocols, and Services Management (PPSM)*, DOD Instruction 8551.01, Department of Defense, Washington, DC, 2014.
- [11] *Cybersecurity*, DOD Instruction 8500.01, Department of Defense, Washington, DC, 2014.

- [12] 2013 top 10 list. (2015, Aug. 26). OWASP. [Online]. Available: [https://www.owasp.org/index.php/Top\\_10\\_2013-Top\\_10](https://www.owasp.org/index.php/Top_10_2013-Top_10).
- [13] M. Howard and S. Lipner, *The Security Development Life Cycle*. Redmond, WA: Microsoft Press, 2006.
- [14] National Institute of Standards and Technology, "Managing information security risk: Organization, mission, and information system view," Special Publication 800–39, March 2011.
- [15] Java Platform, Standard Edition 7 API Specification. (n.d.). Oracle. [Online]. Available: <http://docs.oracle.com/javase/7/docs/api/overview-summary.html>. Accessed Mar. 16, 2015.



## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center  
Ft. Belvoir, Virginia
2. Dudley Knox Library  
Naval Postgraduate School  
Monterey, California